

ERWEITERBARE
OBJEKTERKENNUNGSBASIERTE
AUTOMATISCHE ANNOTATION
VON BILDERN

Der Technischen Fakultät der
Universität Erlangen-Nürnberg

zur Erlangung des Grades

DOKTOR-INGENIEUR

vorgelegt von

Robert Nagy

Erlangen – 2012

Als Dissertation genehmigt von
der Technischen Fakultät der
Universität Erlangen-Nürnberg

Tag der Einreichung: 08.05.2012

Tag der Promotion: 01.08.2012

Dekan: Univ.-Prof. Dr.-Ing. habil. Marion Merklein

Berichterstatter: Univ.-Prof. Dr.-Ing. habil. Klaus Meyer-Wegener
Univ.-Prof. Dr. (ENS Lyon) Harald Kosch

ERKLÄRUNG ZUR SELBSTÄNDIGKEIT

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass diese Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Der Universität Erlangen-Nürnberg, vertreten durch den Lehrstuhl für Informatik 6 (Datenmanagement), wird für Zwecke der Forschung und Lehre ein einfaches, kostenloses, zeitlich und örtlich unbeschränktes Nutzungsrecht an den Arbeitsergebnissen der Dissertation einschließlich etwaiger Schutzrechte und Urheberrechte eingeräumt.

Erlangen, den 08.05.2012

(Robert Nagy)

KURZFASSUNG

Die Flut an digitalen Bildern nimmt Jahr für Jahr stetig zu. Nach einer aktuellen Schätzung werden jährlich 50 Milliarden digitale Fotoaufnahmen erstellt. Um spezifische Bilder auch nur in einem Bruchteil dieser immensen Datenmenge wiederzufinden müssen die Aufnahmen entsprechend indiziert werden. Bei der Bildsuche ist heute die textbasierte Suche am weitesten verbreitet, grafische Verfahren konnten sich bislang nicht durchsetzen. Als Grundlage für die textbasierte Suche dienen manuell oder automatisch erstellte textuelle Annotationen. Die manuelle Indizierung ist bei der enormen Menge an Bildern aussichtslos. Auch die von aktuellen Suchmaschinen eingesetzte Ableitung der Bildbeschreibungen aus dem umliegenden Text von Bildern ist fehleranfällig und ohne entsprechenden – manuell aufwendig erzeugten – Kontext nicht möglich.

Bei der automatischen Annotation von Bildern kann lediglich auf den Inhalt der Bilder sowie evtl. von der Kamera zusätzlich abgespeicherte Metadaten (z. B. Zeit und Ort) zugegriffen werden. In den letzten Jahren wurden wesentliche Fortschritte im Bereich der Objekterkennung gemacht. Diese neuen, auf sog. visuellen Wörtern aufbauenden Objekterkennungsverfahren schneiden zur Zeit am erfolgreichsten bei der inhaltsbasierten Beschreibung von Bildern ab. Allerdings können diese Ansätze nur diejenigen Objekte in den Bildern erkennen, welche auch dem System selbst bekannt sind. Mit der Zeit muss somit das Objekterkennungssystem hinzulernen und um neue Objektklassen erweitert werden. Nahezu alle aktuellen Verfahren sind jedoch bzgl. der Erweiterbarkeit der Objektverzeichnisse stark eingeschränkt, z. T. kann das Hinzufügen einer neuen Objektklasse mehr als 1 Jahr in Anspruch nehmen.

In dieser Arbeit werden die Anforderungen an die textuelle Annotation von Bildern aus Sicht der textbasierten Suche und der Unterstützung von sehbehinderten Personen ermittelt. Anschließend werden die Problemstellen aktueller Verfahren zur Objekterkennung bzgl. der Erweiterbarkeit identifiziert und analysiert. Ausgehend von diesen Erkenntnissen wird eine skalierbare, erweiterbare und möglichst effiziente Lösung ausgearbeitet. Das erstellte neue Verfahren wird unter Verwendung der gängigen Datensätze optimiert und mit erweiterbaren Methoden aus der Literatur verglichen. Zusätzlich werden aktuelle OCR-Werkzeuge daraufhin untersucht inwieweit die Texterkennung in natürlichen Fotoaufnahmen zur Verbesserung der Objekterkennung und der Annotation von Bildern eingesetzt werden kann. Zuletzt wird das erweiterbare Objekterkennungssystem in ein Framework eingebettet, welches als Annotationsdienst für Bildverwaltungsprogramme und textbasierte Suchdienste zur inhaltlichen Beschreibung von Bildern zur Verfügung steht.

ABSTRACT

The sea of digital images is rising constantly from year to year. According to a recent estimate, about 50 billion digital photographs are taken per annum. Retrieving specific images even in a fraction of this massive amount of data is a tough challenge, which requires appropriate indexing of the photographs. Today, the most common method for image search is based on text queries. Graphical approaches could not prevail up to now. The foundations for successful text-based image search are either manually or automatically derived textual annotations. Manual annotation of such a huge number of images is hopeless. Current search engines obtain the annotations for their indexed images from the surrounding text in websites. Unfortunately, these annotations are error-prone and impossible to be derived without the existence of – manually created – context.

Automatic annotation of images relies only on the image content itself and potentially available additional camera metadata such as location or time. Recently, considerable progress has been made in the research area of object recognition. The most successful methods for describing image content are all based on translating so called “visual words” into text. However, all these approaches can only detect objects, which are known by the recognition system. By and by, the recognition system needs to be extended and has to learn new object classes. Almost all current methods are severely limited regarding their extensibility. In some cases, adding a new object class can take up to 1 year.

In this thesis first the requirements for textual annotations of images are analysed from the perspective of text-based image search and the assistance of visually impaired people. Subsequently, current object recognition methods are investigated regarding their extensibility capabilities and the main areas of limitations are identified. Based on these insights, a scalable extensible and preferably efficient solution is developed. This new method is tested and optimised using established datasets. The thesis includes a comparison with other extensible approaches as well. Additionally, current OCR tools are evaluated on their applicability for recognizing text in natural images and their suitability for improving image annotation and object recognition. Finally, the developed extensible object recognition method is embedded in a framework, which can be accessed by image management applications and by text-based search services to annotate images automatically based on the image content only.

INHALTSVERZEICHNIS

1 Einleitung	1
1.1 Ziele	4
1.2 Aufbau der Arbeit	7
2 Begriffs- und Problemdefinition	9
2.1 Eigenschaften	9
2.1.1 Erweiterbarkeit	10
2.1.2 Skalierbarkeit	10
2.1.3 Antwortzeit	11
2.2 Begriffe	12
2.2.1 Bilddaten	12
2.2.2 Mustererkennung	13
2.3 Problemdefinition	18
2.4 Zusammenfassung	20
3 Annotation von Bildern	21
3.1 Motivation für Textannotationen	22
3.1.1 Textbasierte Suche auf Bildern	22
3.1.2 Bilder für Menschen mit Sehbehinderungen	24
3.1.3 Zusammenfassung	27
3.2 Textbasierte Suche auf Bildern	27
3.2.1 Studien zu textuellen Benutzeranfragen auf Bilddatenbanken	28
3.2.2 Logische Strukturierung der Attribute von Bildern	36
3.2.3 Zusammenfassung	46

3.3	Standards für Bildmetadaten und Bildannotationen	47
3.3.1	Standards für Bildmetadaten	47
3.3.2	Konzept-Thesauri für Bildannotationen	50
3.3.3	Zusammenfassung	52
3.4	Zusammenfassung	53
4	Automatische Verfahren zur Bildannotation	55
4.1	Objekterkennung in Bildern	56
4.1.1	Allgemeine Einordnung von Objekterkennungsverfahren	57
4.1.2	Das Bag-of-Words-Konzept	58
4.1.3	Datensätze	77
4.1.4	Bewertungsgrundlagen	80
4.1.5	Wettbewerbe	89
4.1.6	Aktuelle Objekterkennungsverfahren	91
4.1.7	Hierarchien für die Objekterkennung	101
4.1.8	Zusammenfassung	108
4.2	Aktuelle Annotationsverfahren für Bilder	109
4.2.1	Manuelle Annotation	109
4.2.2	Semi-automatische Annotation	112
4.2.3	Vollautomatische Annotation	114
4.2.4	Andere Aufteilungen	118
4.2.5	Zusammenfassung	120
4.3	Zusammenfassung	120
5	Erweiterbare objekterkennungsbasierte Annotation von Bildern	123
5.1	Klassifikation und Merkmale	124
5.1.1	Bag of Words und das visuelle Vokabular	125
5.1.2	Objekte als Szenen	136
5.1.3	Farben	139
5.1.4	Kombination von Merkmalen	141
5.2	Effiziente Suche	144
5.2.1	Indizierung	144
5.2.2	Approximative Nächste-Nachbarn-Suche	147
5.2.3	Komprimierung der Merkmalsvektoren	149
5.2.4	Zusammenfassung	150
5.3	Zusammenfassung	151

6 Optimierung des erweiterbaren Verfahrens	153
6.1 Klassenspezifische visuelle Vokabulare	154
6.1.1 Bestimmung der Vokabulargröße	154
6.1.2 Bestimmung der Schnittgrenze	155
6.1.3 Bestätigung der Outlier-Hypothese	156
6.1.4 Bewertung	156
6.1.5 Zusammenfassung	158
6.2 Szenendeskriptor für Objekte	159
6.2.1 Vergleich mit klassenspezifischen Vokabularen	160
6.2.2 Abhängigkeit von der Anzahl der Trainingsbilder	161
6.2.3 Rotationsinvarianz	162
6.2.4 Unterschiede auf Klassenebene	162
6.2.5 Vollständig automatische Annotation	165
6.2.6 Skalierbarkeit	166
6.2.7 Zusammenfassung	167
6.3 Bag of Colors	168
6.4 Kombination der Merkmale	170
6.5 Einschränkung des Suchraums	171
6.5.1 Durchschnittliche Berechnungszeiten je Merkmal	172
6.5.2 Überlappung der nächsten Nachbarn	173
6.5.3 Auswirkungen auf die Objekterkennung	173
6.5.4 Zusammenfassung	175
6.6 Zusammenfassung	178
7 Architektur des Pixtract-Frameworks	181
7.1 Grobarchitektur von Pixtract	181
7.1.1 Grundkonzepte	182
7.1.2 Lernkomponente	183
7.1.3 Annotationskomponente	183
7.2 Umsetzung	184
7.2.1 Konfigurierbare und erweiterbare Merkmalsextraktion	184
7.2.2 Schnittstellen	185
7.3 Parallelisierung	192
7.4 Säuberung der Stichprobe	195
7.4.1 Filtermechanismus basierend auf dem GIST-Deskriptor	196

7.4.2	Auswirkungen auf die Objekterkennung	198
7.5	Zusammenfassung	201
8	Verbesserung der Annotation durch Text in natürlichen Fotoaufnahmen	203
8.1	Texterkennung in natürlichen Fotoaufnahmen	204
8.1.1	NEOCR Datensatz	205
8.1.2	Globale Metadaten	207
8.1.3	Lokale Metadaten	207
8.1.4	Vergleich mit anderen Datensätzen	212
8.1.5	Distanzmaße für den Vergleich von Zeichensequenzen	217
8.1.6	Evaluation aktueller OCR-Anwendungen	223
8.1.7	Zusammenfassung	238
8.2	Anreicherung der Annotation durch erkannten Text	242
8.3	Zusammenfassung	244
9	Vergleich mit erweiterbaren Ansätzen	247
9.1	Evaluationsaufbau	248
9.2	Klassifikation und Skalierbarkeit	252
9.3	Antwortzeit	254
9.4	Erweiterbarkeit	257
9.5	Zusammenfassung	261
10	Zusammenfassung und Ausblick	263
Anhang		
A	Relevante Felder aus Metadatenstandards	i
B	Linguistische Beziehungen zwischen Wörtern	vii
C	Ähnlichkeitsmetriken für ontologiebasierte Bewertungsmaße	ix
C.1	Kantenbasierte Ähnlichkeitsmaße	ix
C.2	Knotenbasierte Ähnlichkeitsmaße	x
C.3	Hybride Ähnlichkeitsmaße	xi
D	Mathematische Berechnungen	xiii
D.1	Verzerrung	xiii
D.2	Rotation	xviii

E Auflistung von Dateiinhalten	xxi
E.1 XML-Parameterdatei für FeatExt	xxii
E.2 XMP-Ausgabedatei erstellt durch FeatExt	xxiii
E.3 NEOCR XML-Schema-Definition	xxv
E.4 XML-Annotation für ein Beispielbild aus dem NEOCR Datensatz	xxix
F Beispieldaten für den GIST-basierten Filter	xxxi
F.1 Starre Objekte	xxxii
F.2 Bewegliche und verformbare Objekte	xli
F.3 Objekte ohne fester Form	xlv
F.4 Szenen	l
Literaturverzeichnis	lix

ABBILDUNGSVERZEICHNIS

1.1	Schematische Darstellung eines Multimedia Information Retrieval Systems nach [BVBF07].	5
1.2	Eingliederung von Pixtract in die schematische Darstellung des Multimedia Information Retrieval Systems von [BVBF07].	5
2.1	Grundsätzliches Vorgehen bei Klassifikationsproblemen nach [Nie03].	18
3.1	Beispielbilder für die Attributebenen nach [JC02].	42
3.2	Mindmap der verschiedenen Ansätze zur logischen Strukturierung der Bildbeschreibungen.	44
4.1	Strategien zur Klassifikation von Objekten und Szenen nach [BMM07].	58
4.2	Grundsätzliches Vorgehen bei der Erlernung von Objektklassen mittels des Bag-of-Words-Ansatzes.	60
4.3	Grundsätzliches Vorgehen bei der Objekterkennung in Bildern mittels des Bag-of-Words-Ansatzes.	60
4.4	Vergleich von Ecken- und Blob-ähnlichen Detektoren.	61
4.5	Beschreibung von Punkten mit dem SIFT-Deskriptor nach [Low04].	66
4.6	Beispielaufteilungen für Bilder mittels häufig eingesetzter Gitternetze.	68
4.7	Klassifikation der Modelle zur Beschreibung der geometrischen Beziehungen zwischen Punkten nach [CL06].	70
4.8	Linear trennbare Klassen mit optimaler Trenngerade.	72
4.9	Linear nicht trennbare Klassen in \mathbb{R}^2 , deren Abbildung in \mathbb{R}^3 und die optimale Trennfläche nach [MMR ⁺ 01].	73
4.10	Beispiel für die Spatial-Pyramid-Methode nach [BR07].	75

4.11	Darstellung von gängigen Bewertungsmaßen im Bildretrieval und der Objekterkennung.	84
4.12	Allgemeine und klassenspezifische Vokabulare nach [PDCB06].	93
4.13	Bipartite Histogramme und klassenspezifische Klassifikatoren nach [PDCB06].	93
4.14	Objekterkennungsverfahren mit mehreren Kanälen und Farb-SIFT-Deskriptoren nach [TSU ⁺ 08].	95
4.15	Objekterkennung mit einem Vokabularbaum nach [NS06].	102
4.16	Kombinationsmöglichkeiten von SVMs für die Klassifikation in mehrere Klassen.	104
4.17	k -närer SVM-Baum nach [ZWQ ⁺ 05] mit 8 Klassen.	105
4.18	Binärer SVM-Baum nach [GP08] mit 8 Klassen.	106
4.19	DAG-SVM nach [MS08] mit 4 Klassen.	107
4.20	LabelMe: ein webbasiertes Werkzeug zur manuellen Annotation von Bildern.	111
4.21	ALIPR Webseite mit Beispielbild und automatisch ermittelten Tags. . . .	113
4.22	Ablauf der Bildannotation in SADE nach [AYV07].	117
5.1	Deskriptor- und Quantisierungsrauschen in Abhängigkeit von k nach [JDS08].	127
5.2	Beispielhafte visuelle Wörter aus dem visuellen Vokabular für Früchte und interessante Punkte eines Motorradbildes.	128
5.3	Histogramme von visuellen Wörtern für ein Foto einer Weintraube und eines Motorrads unter Verwendung des Früchte-Vokabulars.	128
5.4	Beispiel Objekthierarchie zur top-down bzw. bottom-up Perspektive für den Zusammenhang zwischen dem globalen und den klassenspezifischen Vokabularen.	131
5.5	Berechnung des klassenspezifischen Vokabulars und der Klassenbeschreibung für die Klasse $\Omega_{Blumenkohl}$ anhand einer Stichprobe.	134
5.6	Berechnung des Score-Werts für ein gegebenes Bild I und der Klasse Ω_κ . .	135
5.7	Blickwinkelbasierte Subklassen für die Klasse Fahrrad eingebettet in die Objekthierarchie des PASCAL VOC 2007 Datensatzes.	137
5.8	Beispielbilder der Subklasse Fahrrad _{vorne}	137
5.9	Vertikale Zerlegung einer Anfrage mittels Tree Striping nach [BBK ⁺ 00]. .	146
6.1	Vergleich von verschiedenen Vokabulargrößen $k = 40, 60, 80, 100$ unter Behaltung der Hälfte der häufigsten visuellen Wörter ($n/k = 0,5$).	155

6.2	Vergleich der Genauigkeit der Klassifikation mit unterschiedlicher Anzahl der behaltenen häufigsten visuellen Wörtern n unter Verwendung der Vokabulargröße $k = 100$ für alle klassenspezifische Vokabulare.	156
6.3	Durchschnittliche Anzahl der visuellen Outlier-Wörter für Bilder, welche zur richtigen, und Bilder, welche nicht zur gegebenen Klasse gehören.	157
6.4	ROC-Kurven für Vokabulargröße $k = 100$ mit verschiedenen Werten für die Anzahl der behaltenen häufigsten visuellen Wörter n bei 20 Trainingsbildern je Klasse.	157
6.5	Vergleich der MAP von klassenspezifischen BoW-basierten Ansätzen unter Verwendung von verschiedenen Detektoren bei steigender Anzahl von Klassen.	158
6.6	Vergleich der besten und durchschnittlichen AUC- und MAP-Werte für die verschiedenen GIST-basierten Verfahren mit unterschiedlicher Anzahl von Trainingsbildern.	161
6.7	Durchschnittsbilder erstellt aus 9 Trainingsbildern für die Subklassen <i>rechts</i> , <i>links</i> , <i>vorne</i> und <i>hinten</i> für die Klassen <i>Fahrrad</i> , <i>Motorrad</i> , <i>Katze</i> und <i>Hund</i>	164
6.8	Ergebnisse der vollautomatischen Annotation von 4 ausgewählten Bildern des PASCAL VOC 2007 Datensatzes.	166
6.9	Vergleich der MAP von verschiedenen GIST-basierten Ansätzen bei unterschiedlichen Anzahlen von Klassen.	167
6.10	Vergleich der MAP von verschiedenen Bag of Colors Ansätzen bei unterschiedlichen Anzahlen von Klassen.	169
6.11	Vergleich der MAP für die einzelnen Merkmale und deren kombinierte Verwendung.	170
6.12	Vergleich der eingesetzten Merkmale bzgl. deren Laufzeiten bei der Klassifikation.	172
6.13	Vergleich der Merkmale bzgl. der Überlappung der nächsten Nachbarn mit den Klassifikationsergebnissen der kombinierten Anwendung aller Merkmale.	174
6.14	Beeinflussung der MAP bei der Verwendung einer Vorselektion basierend auf dem GIST-Merkmal in Abhängigkeit der prozentualen Einschränkung des Suchraumes.	176
6.15	Beeinflussung der Laufzeit bei der Verwendung einer Vorselektion basierend auf dem GIST-Merkmal in Abhängigkeit der prozentualen Einschränkung des Suchraumes.	177

Abbildungsverzeichnis

6.16 MAP für die Klassen der obersten Ebenen der Wordnet Hierarchie.	179
6.17 Hierarchischer Fehler für die Klassen der obersten Ebenen der Wordnet Hierarchie.	180
7.1 Grobarchitektur des Pixtract-Frameworks.	182
7.2 Suche und Visualisierung von WordNet Synsets und Klassen in der Pixtract-Webschnittstelle.	186
7.3 Ablauf der Annotation mittels der Pixtract-Webschnittstelle.	188
7.4 Ablauf der Hinzufügung einer neuen Klasse mittels der Pixtract-Webschnittstelle – Teil 1.	189
7.5 Ablauf der Hinzufügung einer neuen Klasse mittels der Pixtract-Webschnittstelle – Teil 2.	190
7.6 Annotation von Bildern eingebettet in die Bildverwaltungsanwendung gThumb.	192
7.7 Ablauf des GIST-basierten Filters zur Eliminierung von Ausreißern und zur Bestimmung der besten Trainingsbilder am Beispiel der Klasse „avocado“.	196
7.8 Vergleich der MAP von der Klassifikation mit ungefilterten, größengefilterten und GIST-gefilterten Trainingsbildern.	200
7.9 Vergleich der MAP der Kombination aller Merkmale mit und ohne Filterung der Trainingsbilder.	200
8.1 Beispiele aus dem NEOCR Datensatz, welche typische Problemfelder der Texterkennung in natürlichen Fotoaufnahmen aufzeigen.	208
8.2 Beispielbild „Finstere Gasse“ in der angepassten Annotationssoftware LabelMe.	214
8.3 Statistiken des NEOCR-Datensatzes bzgl. Helligkeit, Kontrast, Rotation, Abdeckung, Schriftart und Sprache.	215
8.4 Textwahrnehmung durch ausgewählte OCR-Anwendungen in kompletten Bildern mit und ohne Textinhalt.	224
8.5 Beispiele für verzerrte Textausschnitte.	226
8.6 Beispiele für entzerrte Textausschnitte.	226
8.7 Texterkennungsrate für alle Textausschnitte.	227
8.8 Texterkennungsrate für Textausschnitte nach Anordnung gruppiert.	227
8.9 Horizontale Ausschnitte nach Distanzmaß.	228
8.10 Vertikale Ausschnitte nach Distanzmaß.	228
8.11 Zirkulare Ausschnitte nach Distanzmaß.	228

8.12	Texterkennungsrate für Textausschnitte nach Invertierung gruppiert	229
8.13	Texterkennungsrate für Textausschnitte nach Abdeckung gruppiert	230
8.14	Texterkennungsrate für Ausschnitte nach Abdeckungsart (<i>orientation</i>) gruppiert	230
8.15	Texterkennungsrate für Ausschnitte nach prozentualer Abdeckung gruppiert .	231
8.16	Texterkennungsrate für Ausschnitte nach prozentualer Abdeckung und Sprache gruppiert	231
8.17	Texterkennungsrate für Ausschnitte nach Farbe bzw. Textur gruppiert . . .	232
8.18	Texterkennungsrate für Textausschnitte nach Helligkeit gruppiert	232
8.19	Texterkennungsrate für Textausschnitte nach Kontrast gruppiert	233
8.20	Texterkennungsrate für Textausschnitte nach Auflösung gruppiert	234
8.21	Texterkennungsrate für Textausschnitte nach Rauschen gruppiert	234
8.22	Texterkennungsrate für Textausschnitte nach Unschärfe gruppiert	235
8.23	Texterkennungsrate für Textausschnitte nach Verzerrung gruppiert	235
8.24	Texterkennungsrate für Textausschnitte nach Rotation um Vielfache von 90° gruppiert	236
8.25	Texterkennungsrate für Textausschnitte nach Rotation um Winkelbereich bei 0° gruppiert	237
8.26	Texterkennungsrate für Textausschnitte nach Schriftarten gruppiert	237
8.27	Texterkennungsrate für Textausschnitte nach Sprachabhängigkeit gruppiert .	238
8.28	Texterkennungsrate für sprachabhängige Textausschnitte nach Sprachen gruppiert	238
8.29	Texterkennungsrate für sprachunabhängige Textausschnitte nach Katego- rien gruppiert	239
8.30	Beispielbild aus dem ImageNet Datensatz für die Texterkennung und die einzelnen Filterungsschritte	243
9.1	Vergleich der MAP für unterschiedliche Anzahlen von Klassen in Pixtract, sowie mit den Verfahren aus [BSI08] und [AF10].	253
9.2	Vergleich der durchschnittlichen Antwortzeit für die Klassifikation eines Bildes für unterschiedliche Anzahlen von Klassen in Pixtract sowie mit den Verfahren aus [BSI08] und [AF10].	256
9.3	Vergleich der CPU-Zeit für die Aufnahme einer neuen Klasse bei unter- schiedlichen Anzahlen von Trainingsbildern in Pixtract sowie mit den Verfahren aus [BSI08] und [AF10].	259

9.4 Vergleich des durchschnittlich benötigten Speicherplatzes für die Aufnahme einer neuen Klasse bei 20 Trainingsbildern in Pixtract sowie mit den Verfahren aus [BSI08] und [AF10]	260
F.1 Beispiel für beste und schlechteste Bilder für die Kategorie „avocado, alligator pear, avocado pear, aguacate“ (WordNetID: 07764847)	xxxiii
F.2 Beispiel für beste und schlechteste Bilder für die Kategorie „bicycle, bike, wheel, cycle“ (WordNetID: 02834778)	xxxiv
F.3 Beispiel für beste und schlechteste Bilder für die Kategorie „boomerang, throwing stick, throw stick“ (WordNetID: 02871963)	xxxv
F.4 Beispiel für beste und schlechteste Bilder für die Kategorie „brake“ (WordNetID: 02889425)	xxxvi
F.5 Beispiel für beste und schlechteste Bilder für die Kategorie „lemon“ (WordNetID: 07749582)	xxxvii
F.6 Beispiel für beste und schlechteste Bilder für die Kategorie „telephone, phone, telephone set“ (WordNetID: 04401088)	xxxviii
F.7 Beispiel für beste und schlechteste Bilder für die Kategorie „television, television system“ (WordNetID: 04404412)	xxxix
F.8 Beispiel für beste und schlechteste Bilder für die Kategorie „tennis ball“ (WordNetID: 04409515)	xl
F.9 Beispiel für beste und schlechteste Bilder für die Kategorie „kuvasz“ (WordNetID: 02104029)	xlii
F.10 Beispiel für beste und schlechteste Bilder für die Kategorie „paper“ (WordNetID: 06255613)	xliii
F.11 Beispiel für beste und schlechteste Bilder für die Kategorie „towel“ (WordNetID: 04459362)	xliv
F.12 Beispiel für beste und schlechteste Bilder für die Kategorie „butter“ (WordNetID: 07848338)	xlvi
F.13 Beispiel für beste und schlechteste Bilder für die Kategorie „double creme, heavy whipping creme“ (WordNetID: 07847585)	xlvii
F.14 Beispiel für beste und schlechteste Bilder für die Kategorie „milk“ (WordNetID: 07844042)	xlviii
F.15 Beispiel für beste und schlechteste Bilder für die Kategorie „paper“ (WordNetID: 14974264)	xlix

F.16 Beispiel für beste und schlechteste Bilder für die Kategorie „billiard room, billiard saloon, billiard parlor, billiard parlour, billiard hall“ (WordNetID: 02839592).	li
F.17 Beispiel für beste und schlechteste Bilder für die Kategorie „bite, collation, snack“ (WordNetID: 07577374).	lii
F.18 Beispiel für beste und schlechteste Bilder für die Kategorie „open-air market, open-air marketplace, market square“ (WordNetID: 03847823). .	liii
F.19 Beispiel für beste und schlechteste Bilder für die Kategorie „pasta“ (WordNetID: 07863374).	liv

TABELLENVERZEICHNIS

3.1	Kategorisierung der Veröffentlichungen zur textbasierten Suche nach Bildern – Teil 1.	37
3.2	Kategorisierung der Veröffentlichungen zur textbasierten Suche nach Bildern – Teil 2.	38
3.3	Panofsky-Shatford Matrix nach [AE97].	40
3.4	Vergleichende Tabelle der verschiedenen Ansätze zur logischen Strukturierung der Bildbeschreibungen.	45
3.5	Vergleich von verschiedenen Konzept-Thesauri für die Annotation von visuellen Medien.	52
4.1	Überblick über Detektoren für interessante Punkte in Bildern nach [MTS ⁺ 05] und [TM08].	62
4.2	Überblick über Deskriptoren für interessante Punkte in Bildern.	65
4.3	Überblick über Distanzmaße in Objekterkennungsverfahren.	69
4.4	Vergleich von verschiedenen Datensätzen für die Evaluation von Objekterkennungsverfahren.	80
4.5	Entwicklung der PASCAL VOC Datensätze für die Objekterkennungsaufgabe.	89
4.6	Vergleich der Objekterkennungswettbewerbe PASCAL VOC, ImageCLEF und ILSVRC.	90
4.7	Vergleich von den besten Objekterkennungsansätzen – Teil 1.	99
4.8	Vergleich von den besten Objekterkennungsansätzen – Teil 2.	100
4.9	Aufteilung der Veröffentlichungen zur Annotation von Bildern mittels verschiedener Kategorisierungen.	119

Tabellenverzeichnis

6.1	Vergleich der AUC des erweiterbaren BoW-Ansatzes mit den GIST-basierten Ansätzen.	160
6.2	Vergleich der AP der besten und Durchschnitts-AP-Werte des PASCAL VOC 2007 Wettbewerbs mit dem vorgestellten meanGIST- und NNGIST-Ansatz.	163
6.3	Übersicht der MAP für die einzelnen Merkmale und deren Kombination bei 3 251 Klassen.	171
8.1	Unterschiede zwischen Text in eingescannten Dokumenten und Text in natürlichen Fotoaufnahmen.	206
8.2	Datentypen und Wertebereiche der Metadaten sowie Beispiel-Annotationswerte für Abbildung 8.2.	213
8.3	Vergleich verschiedener Datensätze für die Texterkennung in natürlichen Fotoaufnahmen.	217
8.4	Übersicht untersuchter Eigenschaften natürlicher Fotoaufnahmen mit jeweiligem Optimum.	241
A.1	Bildbeschreibende Attribute aus dem IPTC Extension Schema [IPT08].	ii
A.2	Attribute des Datentyps ArtworkOrObjectDetails nach [IPT08].	iii
A.3	Attribute des Datentyps LocationDetails nach [IPT08].	iii
A.4	Bildbeschreibende Attribute aus den XMP Schemata – Teil 1.	iv
A.5	Bildbeschreibende Attribute aus den XMP Schemata – Teil 2.	v
B.1	Semantische Beziehungen zwischen Wörtern nach [Mil95].	viii

LISTINGS

4.1	SIFT-Deskriptoren für mehrere Punkte.	66
8.1	Der durch Tesseract OCR erkannte Text für Abbildung 8.30 im Rohformat.	243
8.2	Ergebnis nach Filterung der Sonderzeichen und Entfernung kurzer Zeichenketten.	243
8.3	Ergebnis nach der Lemmatisierung.	244
8.4	Endergebnis nach WordNet-Filterung.	244
E.1	XML-Parameterdatei für die Konfiguration von <i>FeatExt</i>	xxii
E.2	XMP-Ausgabedatei erstellt durch <i>FeatExt</i>	xxiii
E.3	XML-Schema für den NEOCR Datensatz.	xxv
E.4	XML-Annotation für Abbildung 8.2 auf Seite 214.	xxix

ABKÜRZUNGSVERZEICHNIS

ALIP	Automatic Linguistic Indexing of Pictures
ALIPR	Automatic Linguistic Indexing of Pictures - Real-Time
ANN	approximative nächste Nachbarn
AP	Average Precision
AUC	Area Under the ROC Curve
BLOB	Binary Large Object
BoC	Bag of Colors
BoW	Bag of Words
CIPA	Camera & Imaging Products Association
CLEF	Cross Language Evaluation Forum
COIL	Columbia Object Image Libraries
CMRM	Cross-Media Relevance Model
CRF	Conditional Random Fields
CRM	Continuous-space Relevance Model
DAG	Directed Acyclic Graph
DCT	Discrete Cosine Transform
DISC	Digital Image Submission Criteria
DoG	Difference of Gaussian

Abkürzungsverzeichnis

DP	Data Partitioning
EBU	European Broadcasting Union
EM	Expectation-Maximization Algorithmus
EMD	Earth-Mover Distanz
ESP	Extra Sensory Perception
Exif	Exchangeable image file format
GB	Geometric Blur Deskriptor
GFS	Google File System
GiST	Generalized Search Tree
GIST	GIST Merkmal
GLOH	Gradient Location-Orientation Histogram
GMM	Gaussian Mixture Model
HDFS	Hadoop Distributed File System
HDP	Hierarchical Dirichlet Process
HoG	Histogram of Gradients
HSV	Hue Saturation Value (Farbraum)
IAPR	International Association of Pattern Recognition
IDC	International Data Corporation
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
IPTC	International Press Telecommunications Council
IPTC-IIM	IPTC Information Interchange Model
ISO	International Organization for Standardization
JAWS	Job Access With Speech
JEITA	Japan Electronics and Information Technology Industries Association
JPEG	Joint Photographic Experts Group
kAS	k Adjacent Segments
KDA	Kernel Discriminant Analysis

kNN	k-nächste Nachbarn
LAMPP	Linux, Apache, MySQL, PHP, Perl
LCS	Longest Common Substring
LDA	Latent Dirichlet Allocation
LoG	Laplacian of Gaussian
LSH	Locality Sensitive Hashing
LSCOM	Large-Scale Concept Ontology for Multimedia
MAP	Mean Average Precision
MAWG	Media Annotations Working Group
MBRM	Multiple Bernoulli Relevance Modell
MCMC	Markov Chain Monte Carlo Algorithmus
MIRS	Multimedia Information Retrieval System
MKL	Multiple Kernel Learning
MLE	Maximum Likelihood Estimation
MPI	Message Passing Interface
MSER	Maximally Stable Extremal Region
MSRC	Microsoft Research Cambridge
NBNN	Naive-Bayes nächster Nachbar (Klassifikator)
NCRP	Nested Chinese Restaurant Process
NEOCR	Natural Environment OCR
NLTK	Natural Language Toolkit
NN	nächste Nachbarn
ObjectFP	Object Fingerprint
OCR	Optical Character Recognition
OpenMP	Open Multi Processing
OWL	Web Ontology Language
PAS	Pairs of Adjacent Segments

PASCAL	Pattern Analysis, Statistical modelling, Computational Learning
PASCAL VOC	PASCAL Visual Object Class Challenge
PCA	Principal Component Analysis
PHOG	Pyramid Histogram of Oriented Gradients
PHOW	Pyramid Histogram of Words
Pixtract	Picture Annotation Extraction
pLSA	Probabilistic Latent Semantic Analysis
PNG	Portable Network Graphics
POSIX	Portable Operating System Interface
P-RBF	Pyramid Radial Basis Function
PRISM	Publishing Requirements for Industry Standard Metadata
QBIC	Query by Image Content
RBF	Radial Basis Function
RDF	Resource Description Framework
REST	Representational State Transfer
RGB	Rot Grün Blau (Farbraum)
RIFT	Rotation-Invariant Feature Transform
ROC	Reciever Operating Characteristic
ROI	Region of Interest
SADE	Supervised Annotation by Descriptor Ensemble
SIFT	Scale-Invariant Feature Transform
SKM	Support Kernel Machine
SP	Space Partitioning
SRKDA	Spectral Regression Kernel Discriminant Analysis
SURF	Speeded-Up Robust Features
SUSAN	Smallest Univalue Segment Assimilating Nucleus
SVM	Support Vector Machine

TIFF	Tagged Image File Format
TREC	Text REtrieval Conference
TRECVID	TREC Video Retrieval Evaluation
UCID	Uncompressed Color Image Database
W3C	World Wide Web Consortium
WCAG	Web Content Accessibility Guidelines
WHO	World Health Organization
XML	Extensible Markup Language
XMP	Extensible Metadata Platform
XRCE	Xerox Research Center Europe
YUI	Yahoo! User Interface Library

KAPITEL 1

EINLEITUNG

„Nichts ist im Verstand, was nicht vorher in den Sinnen gewesen wäre.“

John Locke [Loc90]

Die Menschheit befindet sich im digitalen Zeitalter. Immer mehr Information wird digitalisiert, oder sogar nur noch digital erstellt und abgespeichert. Die Verbreitung von digitalen Verfahren, die immer breitere Vielfalt an digitalen Geräten und die billigen Speichermedien haben zu einer rapiden Vergrößerung der Datenmengen geführt.

Die Statistiken für das Ausmaß an digitalen Bildern sind enorm: Nach den Daten der Camera & Imaging Products Association (CIPA) wurden im Jahr 2011 weltweit 115,5 Millionen Digitalkameras ausgeliefert [Ass11]. Zusätzlich kommen nach einer Schätzung der International Data Corporation (IDC)¹ für das Jahr 2011 472 Millionen Smartphones hinzu, die ebenfalls als Aufnahmegeräte für digitale Bilder verwendet werden. Einer der größten Foto-Gemeinschaften des Internets, Flickr² verzeichnet nach [Sap11] 5 000 neue Bilder pro Minute und insgesamt 7 Millionen am Tag. Die bei weitem größte

1 <http://www.idc.com/getdoc.jsp?containerId=prUS22871611>

2 <http://www.flickr.com>

1. Einleitung

Online-Fotosammlung der Welt ist auf Facebook zu finden. Laut [Sav11] wurden in 2010 ca. 30 Milliarden Bilder hochgeladen, aktuell beläuft sich der Zuwachs im Durchschnitt auf 250 Millionen Bilder pro Tag¹. Umgerechnet auf ein Jahr sind das ca. 90 Milliarden Bilder, was sogar die aktuelle Schätzung von 50 Milliarden erstellten Bilder pro Jahr aus [SS11] übertrifft.

Auch wenn nicht alle knapp 100 Milliarden Bilder pro Jahr archiviert und zu einer späteren Zeit wiedergefunden werden müssen, stellt bereits ein Bruchteil dieser enormen Menge ein erhebliches Problem dar. Um zu einem späteren Zeitpunkt in diesen riesigen Datenbergen relevante Dokumente in relativ kurzer Zeit wiederzufinden sind einerseits gute Suchstrategien und Indizierungstechniken, andererseits aber auch die Anreicherung der Dokumente durch entsprechende Metadaten notwendig.

Bilder können in ihrem Rohformat nur schlecht durchsucht werden, da die Bildpunkte alleine keine höheren semantischen Konzepte, wie z. B. Autos oder Brücken beschreiben. Diese Merkmale auf hoher Ebene (z. B. Objekte, Personen, Ereignisse) müssen aus Merkmalen der niedrigeren Stufe (z. B. Farben, Formen, Texturen) durch spezielle Algorithmen oder manuell ermittelt und zum Bild annotiert werden. Die manuelle Annotation hat jedoch die Nachteile, dass sie sowohl zeit- als auch kostenintensiv ist. Außerdem können die Annotationen mehr oder weniger ausführlich ausfallen sowie von unterschiedlichen Personen verschiedene Begriffe für das gleiche Objekt verwendet werden. Sofern zur Behebung dieser Abweichungen nichts unternommen wird, können diese Eigenschaften die spätere Suche nach Bildern negativ beeinflussen.

Die am weitesten verbreitete Suchmethode im Web ist die textbasierte Suche. Suchdienste wie z. B. Google², Yahoo!³ oder bing⁴ bieten nur diese Suchoption an, aber auch bei speziellen Bildarchiven wie z. B. Flickr oder iStockphoto⁵ ist nur die Möglichkeit basierend auf Schlüsselwörtern nach Bildern zu suchen gegeben. Neben der textbasierten Suche existieren auch Suchverfahren, bei denen nach Farben, Formen, Texturen oder durch die Angabe eines Beispielbildes gesucht werden kann. Für den Benutzer kann es jedoch oft schwierig sein ein Beispielbild anzugeben, wenn er eben gerade auf der Suche nach einem Beispielbild ist. Farben, Formen und Texturen können in Situationen, bei

1 <http://www.facebook.com/press/info.php?statistics>

2 <http://www.google.com>

3 <http://www.yahoo.com>

4 <http://www.bing.com>

5 <http://www.istockphoto.com>

denen man diese Merkmale mit Worten nur schwer spezifizieren kann, behilflich sein. Diese Merkmale werden jedoch bei der Suche im Web kaum verwendet, da der Benutzer oft keine genaue strukturelle Vorstellung vom gesuchten Bild hat.

Am häufigsten wird nach Merkmalen der höheren Ebene (Objekte, Personen, Ereignisse) gesucht. Diese Konzepte sind auch textuell beschreibbar, sofern sie vorab von Personen oder entsprechenden Algorithmen erkannt und zum Bild annotiert wurden. Sie können also durch die textbasierte Suche gut angefragt werden. Zusätzlich ist die textbasierte Suche intuitiver und benutzerfreundlicher, da keine speziellen Werkzeuge zur Anfrage benötigt werden und man sich im Alltag auch meistens mittels Worten ausdrückt. Die textbasierte Suche ist auch transparenter, da z. B. eine Farbe allein noch kein Objekt identifizieren kann, ein Konzept auf einer höheren Ebene jedoch schon. Somit kann auch besser nachvollzogen werden, warum ein Bild als Ergebnis zu einer gegebenen Anfrage bei der textbasierten Suche auftaucht bzw. nicht zurückgeliefert wird.

Um in Bildern textbasiert suchen zu können werden von den großen Suchdiensten Schlüsselwörter aus den umliegenden Texten bzw. aus der Webseite extrahiert und mit dem Bild verbunden. Eine genauere, treffendere Beschreibung der Bilder ist durch die Verwendung von speziellen HTML-Tags (z. B. `alt`, `title`) möglich, erfordert heute aber meistens weiterhin manuellen Zusatzaufwand bei der Erstellung der Annotation. Zwar existieren Systeme die Bilder inhaltsbasiert indizieren, jedoch verwendet keines der großen Suchdienste (Google, bing, Flickr u. Ä.) dieses Verfahren. Das heißt, es werden heute noch keine automatischen Verfahren zur Ermittlung von Objekten aus Bildinhalten von Suchdiensten im großen Umfang verwendet. Begründet werden kann dies durch Schwierigkeiten bei der automatischen Bildsegmentierung, Ungenauigkeiten bei der Erkennung von Objekten in Bildern, fehlende Skalierbarkeit der Verfahren auf mehrere 100, 1 000 oder 10 000 Klassen sowie die größtenteils nicht vorhandene Erweiterbarkeit aktueller Ansätze durch neue Klassen.

Die Anzahl der Digitalbilder nimmt mit der Verbreitung von Digitalkameras, billigen Speichermedien und der Digitalisierung von kunsthistorischen bzw. Bibliothekssammlungen stetig zu, wodurch eine manuelle Anreicherung der Bilder mit Beschreibungsinformationen unmöglich ist. Im letzten Jahrzehnt sind verschiedene Ansätze zur automatischen Erkennung von Objekten bzw. Personen in Bildern entstanden, die mehr oder weniger generisch ausgerichtet sind. Die meisten Ansätze können einige 10 bis 100 verschiedene Objekte in Bildern erkennen, wobei die Genauigkeit der Erkennung stark je nach Verfahren, Einsatzgebiet und beabsichtigtem Rechenaufwand variiert.

Nach einer Schätzung von [Bie87] kann ein Mensch visuell auf Anhieb 30 000 unterschiedliche Objekte identifizieren. Die Skalierung der Objekterkennungsansätze auf eine so hohe Anzahl von verschiedenen Objekten (bzw. Klassen) ist zur Zeit ein offenes Problem. Auch die einfache Erweiterbarkeit von Objekterkennungssystemen ist bislang ungelöst. Beim Hinzufügen einer neuen Klasse muss bei aktuellen Ansätzen das ganze Erkennungssystem vollständig neu aufgesetzt werden. Nach [DBLFF10] kann dieser „Neustart“ je nach Verfahren bereits bei 10 000 Klassen 1 bis 100 Jahre dauern – und mit der steigenden Zahl von Klassen nimmt die benötigte Rechenzeit weiter zu.

Ausgehend von den aktuell besten Verfahren zur Objekterkennung und Annotation ist somit eine Lösung zu finden, die skalierbar, erweiterbar und möglichst effizient ist. Dadurch wäre ein weiterer Schritt für die inhaltsbasierte Beschreibung von Bildern in die Richtung von bereits etablierten textbasierten Web-Suchdiensten gemacht. Diese erlauben dank Jahrzehntelanger Forschung das schnelle Auffinden von Texten bzw. Webseiten und ermöglichen zusätzlich die dynamische Erweiterung ihrer Datenbestände.

1.1 Ziele

In dieser Arbeit wird eine erweiterbare automatische inhaltsbasierte Annotation von Bildern realisiert. Die ermittelten Annotationen sollen später zur textbasierten Suche in Bildern dienen. Dabei werden die merkmals- und die textbasierte Suche voneinander getrennt. Ersteres wird für die Annotation von Bildern verwendet und im Picture Annotation Extraction (Pixtract) Framework realisiert. Auf den erstellten Annotationen kann anschließend mit bereits etablierten Methoden der Textindizierung ein Suchdienst aufgebaut werden.

Abbildung 1.1 veranschaulicht den schematischen Aufbau eines Multimedia Information Retrieval System (MIRS) nach [BVBF07]. Das Hauptaugenmerk dieser Arbeit liegt auf der Komponente, welche für die Merkmalsextraktion zuständig ist. Diese wird in ein getrenntes System (Pixtract) ausgelagert. Die Eingliederung dieser Arbeit in den Aufbau eines MIRS ist in Abbildung 1.2 dargestellt.

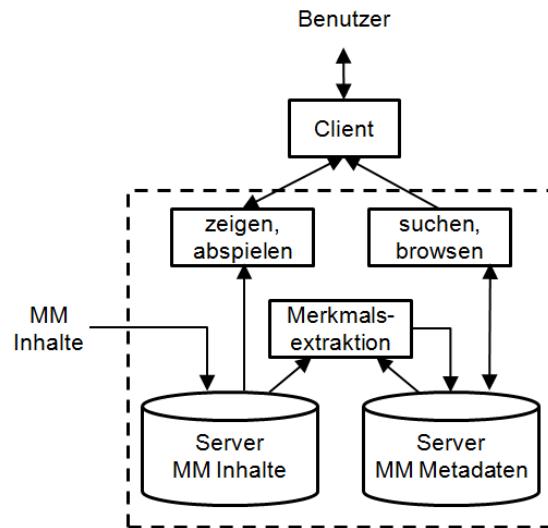


Bild 1.1: Schematische Darstellung eines Multimedia Information Retrieval Systems nach [BVBF07].

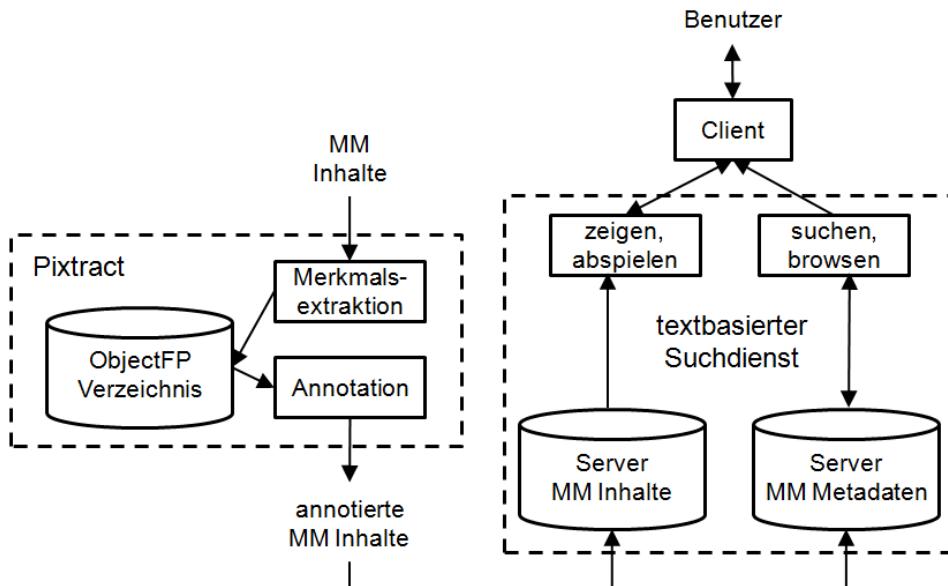


Bild 1.2: Eingliederung von Pixtract in die schematische Darstellung des Multimedia Information Retrieval Systems von [BVBF07]. Diese Abbildung veranschaulicht das Pixtract Framework nur in einer stark eingeschränkten Form. Auf den detaillierten Aufbau von Pixtract wird in Kapitel 7 ausführlich eingegangen.

1. Einleitung

Im Rahmen dieser Dissertation werden folgende Beiträge geleistet:

- *Ermittlung der Anforderungen an die textuelle Annotation von Bildern.*
Viele Arbeiten auf dem Gebiet der inhaltsbasierten Bildannotation gehen direkt auf die Erkennung von Objekten, Personen oder Szenen ein, ohne vorher zu untersuchen, was überhaupt für das jeweilige Anwendungsgebiet annotiert werden soll. Ausgehend von Untersuchungen und Auswertungen der Literatur werden in dieser Arbeit die Anforderungen an Bildannotationen sowohl aus der Sicht der textbasierten Bildsuche, als auch für Menschen mit Sehbehinderungen definiert.
- *Identifikation von Problemstellen bzgl. der Erweiterbarkeit von Objekterkennungsansätzen und Erarbeitung einer erweiterbaren Lösung.*
Aktuelle Objekterkennungsverfahren können in den meisten Fällen nicht einfach durch neue Klassen erweitert werden. In dieser Arbeit werden die Gründe für dieses Problem auf verschiedenen Ebenen des Erkennungsvorgangs identifiziert und ausführlich diskutiert. Basierend auf diesen Erkenntnissen wird eine erweiterbare Lösung erarbeitet. Da die Bilder für neue Klassen von den Anwendern stammen, wird ein Säuberungsmechanismus entworfen, welcher stark abweichende Bilder ausselektiert. Zusätzlich wird auch die Skalierbarkeit der erweiterbaren Lösung gewährleistet, d. h. auch bei mehreren 1 000 Klassen nimmt die Ermittlung der Annotation nicht zu viel Zeit in Anspruch und ist im Bereich des Stands der Technik.
- *Evaluation der Texterkennung in Bildern für die Verbesserung der Annotation.*
Zur Verbesserung der Objekterkennung und zur Anreicherung der Annotation wird der aktuelle Stand der Texterkennung in Bildern untersucht. Für die Evaluation werden die wesentlichen Unterschiede zwischen natürlichen Fotoaufnahmen und eingescannten Dokumenten identifiziert. Basierend darauf werden die Metadaten definiert, anhand derer ein neu erstellter konfigurierbarer Datensatz annotiert wird. Dank der detaillierten Auswertung mittels des Natural Environment OCR (NEOCR) Datensatzes werden wesentliche Schwachstellen aktueller OCR-Werkzeuge ermittelt. Der NEOCR Datensatz ist frei verfügbar und dient durch die Annahme der IAPR-TC11¹ „Reading Systems“ der OCR-Community zur Verbesserung der Texterkennung in natürlichen Fotoaufnahmen.

1 <http://www.iapr-tc11.org>

1.2 Aufbau der Arbeit

Diese Arbeit gliedert sich in zehn Kapitel.

In Kapitel 2 werden die verwendeten Grundbegriffe definiert. Für die Erstellung eines leicht erweiterbaren Systems zur Annotation von Bildern werden in Kapitel 3 zuerst die Anforderungen an die textuelle Annotation untersucht. Dabei wird in Abschnitt 3.1 vor allem der Nutzen der textuellen Annotation von Bildern in verschiedenen Bereichen hervorgehoben. Das Hauptaugenmerk liegt auf der Verwendung von Bildern in der textbasierten Suche, somit werden in Abschnitt 3.2 bislang durchgeführte Untersuchungen zur Annotation von Bildern sowie Strukturierungsansätze für Bildannotationen vorgestellt. In Abschnitt 3.3 werden die am weitesten verbreiteten Metadatenstandards für Bilder kurz präsentiert. Als Ausgangspunkt für die erweiterbare Annotation von Bildern dienen aktuelle Verfahren der Objekterkennung, welche in Kapitel 4 vorgestellt werden. Abschnitt 4.1 führt in das Vorgehen der allgemeinen Objekterkennung in Bildern ein und stellt die besten Ansätze aus der Literatur vor. Anschließend werden in Abschnitt 4.2 Arbeiten zu manuellen, semi-automatischen und vollautomatischen Annotationsverfahren präsentiert.

Ausgehend von den Erkenntnissen aus Kapitel 4 werden in Kapitel 5 wesentliche Problemstellen bzgl. der Erweiterbarkeit von Objekterkennungsansätzen identifiziert und eine erweiterbare Lösung für die Annotation von Bildern erarbeitet. Die Optimierung des erweiterbaren Verfahrens wird in Kapitel 6 durchgeführt. Die Lösung wird anschließend im Pixtract Framework umgesetzt, welches in Kapitel 7 im Detail vorgestellt wird. Zur Verbesserung der Klassifikation und der Annotation durch die Erkennung von Text in Bildern werden in Kapitel 8 aktuelle OCR-Werkzeuge anhand des NEOCR Datensatzes evaluiert. Im Anschluss wird in Kapitel 9 die vorgestellte Lösung mit anderen erweiterbaren Ansätzen verglichen.

Kapitel 10 blickt auf die erzielten Ergebnisse der vorliegenden Arbeit zurück und diskutiert weitere Möglichkeiten zur Verbesserung der Annotation von Bildern.

KAPITEL 2

BEGRIFFS- UND PROBLEMDEFINITION

Dieses Kapitel definiert die grundlegende Terminologie, die in der vorliegenden Arbeit verwendet wird, und grenzt anschließend aufbauend auf den Begriffsdefinitionen die zu lösenden Probleme ab.

Als Erstes werden in Abschnitt 2.1 wesentliche Eigenschaften des zu erstellenden Systems definiert. Abschnitt 2.2 führt Begriffe für die Klassifikation ein. Aufbauend auf den Eigenschaften und Begriffen werden in Abschnitt 2.3 die Ziele dieser Arbeit genau definiert und abgegrenzt.

2.1 Eigenschaften

Die Begriffe Erweiterbarkeit und Skalierbarkeit werden sowohl in verschiedenen als auch in den selben Bereichen der Informatik unterschiedlich verwendet. Um eine einheitliche Basis für diese Arbeit zu schaffen, werden nachfolgend die unterschiedlichen Begriffserklärungen kurz erläutert und eine eindeutige Definition abgeleitet.

2.1.1 Erweiterbarkeit

Der Begriff der Erweiterbarkeit wird innerhalb der Informatik in unterschiedlichen Bereichen verwendet. [Mös94] trennt Erweiterbarkeit auf der Ebene von Programmiersprachen und auf Systemebene. Erweiterbare Programmiersprachen ermöglichen das Hinzufügen von neuen Operationen und Datentypen zu einem Programm, ohne dabei bereits bestehenden Code ungültig zu machen. Auf Systemebene müssen bereits geladene Programme durch neue Module mit neuen Datentypen und Programmcode erweiterbar sein, wobei die neuen Module zum gleichen Addressraum zugeteilt werden sollten wie das geladene Programm, um dessen Daten und Funktionen ansprechen zu können. Im Software Engineering steht der Begriff nach [Raa93, Kah01] für die Einfügung von neuen Objekten oder Funktionen in ein Softwaresystem, ohne dabei Letzteres wesentlich verändern zu müssen. In den meisten Fällen wird unter diesen Veränderungen die Anpassung von existierendem Code verstanden.

Basierend auf diesen Definitionen wird in der vorliegenden Arbeit unter Erweiterbarkeit folgendes Ziel verfolgt: Die Erweiterung des zu erstellenden Objekterkennungs- und Annotationssystems durch neue Klassen soll ohne wesentliche Veränderungen und ohne lange Verzögerungen möglich sein. Verzögerungen ergeben sich bei der Erweiterung der Klassenmenge insbesondere durch die ggf. erforderliche Neuberechnung von Merkmalen bzw. erneutes Training von Klassifikatoren. Diese Faktoren gilt es im nachfolgenden Teil der Arbeit zu vermeiden bzw. einzuschränken, ohne dabei die Genauigkeit der Objekterkennung und der damit verbundenen Annotation maßgeblich zu verringern.

Zusätzlich wird auch auf die leichte Erweiterbarkeit von Pixtract durch neue Merkmale geachtet. Dadurch soll es ermöglicht werden zukünftige ggf. besser geeignete Merkmale einfach und schnell in Pixtract aufzunehmen, ohne dass dabei das Framework bzw. bereits vorhandene Merkmale und Klassifikatoren beeinträchtigt werden.

2.1.2 Skalierbarkeit

Eng gekoppelt mit der Erweiterbarkeit ist die Skalierbarkeit von Systemen. Der Begriff findet innerhalb der Informatik eine sehr breite Verwendung, weswegen eine eindeutige und saubere Definition leider nicht möglich ist. Mehrere Veröffentlichungen haben bereits versucht den Begriff der Skalierbarkeit genau einzugrenzen [Hil90, Luk93, Gus94, Bon00, BH04], jedoch ohne Erfolg. Aus einem sehr allgemeinem Betrachtungswinkel definiert

[DRW06] Skalierbarkeit als die Fähigkeit eines Systems sich zur „Skalierung“ bzgl. einer oder mehrerer Dimensionen anzupassen. Skalierbarkeit kann somit nur im Kontext der Anforderungen an das System genauer eingeschränkt und definiert werden.

In der Objekterkennung und Bildannotation wird der Begriff der Skalierbarkeit für eine möglichst gute „Skalierung“ auf eine hohe Anzahl von Bildern oder Klassen verwendet [NS06, DJLW08, WJZZ08]. Hierbei wird unter „Skalierung“ sowohl die Möglichkeit zur Indizierung von großen Bildmengen als auch eine möglichst gute Erkennungsrate bei einer hohen Anzahl von Bildern oder Klassen verstanden. Die erwähnte hohe Anzahl selbst wird unterschiedlich aufgefasst und variiert von 100 bis 10 000 Klassen. Eine dynamische Erweiterbarkeit durch neue Bilder oder Klassen sowie eine schnelle Antwortzeit des Systems wird bei der Skalierbarkeit in diesem Umfeld allerdings nicht gefordert. Sinngemäß wird in der vorliegenden Arbeit der Begriff der Skalierbarkeit als eine möglichst gute Erkennungsrate bei einer hohen Anzahl von vorhandenen Klassen im System verstanden.

2.1.3 Antwortzeit

Eine weitere wesentliche Eigenschaft von Systemen ist eine möglichst kurze Antwortzeit. Allgemein wird der Begriff der Antwortzeit nach [Mey07] definiert als die Zeit zwischen dem Auslösen einer Anfrage und dem vollständigen Eintreffen der zugehörigen Antwort.

Im Rahmen der vorliegenden Arbeit wird die Antwortzeit unter verschiedenen Aspekten beleuchtet. Einerseits wird versucht die Erweiterbarkeit des Systems so zu optimieren, dass bei der Hinzufügung einer neuen Klasse das System in einer möglichst kurzen Zeit wieder einsatzbereit ist. In diesem Sinne wird unter der Antwortzeit die Zeit zwischen der Übermittlung der Trainingsbilder und dem Abschluss des Erlernens der neuen Klasse durch das Objekterkennungssystem betrachtet.

Aus Sicht der Annotation eines Bildes bezieht sich die Antwortzeit hingegen auf die Zeit zwischen der Übermittlung eines Bildes und dem vollständigen Eintreffen der zugehörigen Annotation. Bei verschiedenen Optimierungsentscheidungen werden zum Teil auch lediglich die Laufzeiten von Merkmalsvergleichen untersucht, ohne eine vollständige Annotation zu erstellen.

Zuletzt wird unter der Antwortzeit aus Sicht der textbasierten Suche die Zeit zwischen der Absendung einer Anfrage und der Rückgabe von relevanten Bildern verstanden. Da im Rahmen dieser Arbeit für die textbasierte Indizierung der automatisch annotierten

Bilder externe Suchdienste verwendet werden, wird diese Antwortzeit in der Arbeit nicht weiter untersucht.

2.2 Begriffe

In diesem Abschnitt werden die wesentlichen Begriffe dieser Arbeit definiert. Die verschiedenen Ebenen für Bilddaten werden in Abschnitt 2.2.1 vorgestellt. Abschnitt 2.2.2 definiert die Terminologie der Klassifikation von Bildern.

2.2.1 Bilddaten

Nach [MW03] können für Medienobjekte Roh-, Registrierungs- und Beschreibungsdaten unterschieden werden. Diese werden nachfolgend kurz definiert.

Rohdaten

Rohdaten sind eine Folge von Bits, die als unformatierte Daten das Medienobjekt selbst beinhalten. Bei Bildern bestehen die Rohdaten z. B. aus den einzelnen Bildpunkten (Pixeln) mit den zugehörigen Farbwerten. Ohne zusätzliche Daten, welche den Kontext für die Interpretation der Rohdaten angeben, ist eine Darstellung der Bilder nur bedingt möglich.

Registrierungsdaten

Registrierungsdaten dienen der korrekten Interpretation der Rohdaten. Bei Bildern sind diese zusätzlichen Daten meistens in den Headern der Dateien vermerkt und geben z. B. die Höhe und Breite des Bildes oder die Pixeltiefe an. Registrierungsdaten werden weiterhin durch identifizierende Daten ergänzt, welche zusätzliche Angaben (wie z. B. Zeitpunkt und Ort der Aufnahme) abspeichern. Für gewöhnlich stehen diese Informationen in den Exif-Daten der Bilder.

Beschreibungsdaten

Beschreibungsdaten charakterisieren den Inhalt oder die Struktur von Medienobjekten. In erster Linie dienen diese Beschreibungsdaten dem einfacheren und schnelleren Vergleich

von Medienobjekten und der damit verbundenen Suche. Bilder können z. B. durch Merkmale wie Farben, Formen und Texturen charakterisiert werden, von denen Beschreibungen wie Objekte und Personen im Bild abgeleitet werden können. Da Letztere automatisch nur schwierig aus dem Bildinhalt zu ermitteln sind, werden diese Informationen häufig manuell oder semi-automatisch als Text zu den Bildern annotiert.

2.2.2 Mustererkennung

Nachfolgend wird die Terminologie bzgl. der Klassifikation von Bildern für die weiteren Kapitel nach [Nie83], [Nie03], [DHS00] und [DJLW08] definiert. Nach jedem Begriff wird der konkrete Bezug zur Objekterkennung bzw. Annotation anhand von Beispielen veranschaulicht.

Umwelt

Gegenstand der Wahrnehmung sind Eindrücke aus der Umwelt. Die Umwelt ist die Gesamtheit der physikalisch messbaren Größen, welche durch die Menge in Gleichung 2.1 formal definiert ist, wobei ${}^{\rho}\mathbf{b}(\mathbf{x})$ messbare Größen oder Funktionen darstellen.

$$U = \{{}^{\rho}\mathbf{b}(\mathbf{x}) | \rho = 1, 2, \dots\} \quad (2.1)$$

Problemkreis

Die Umwelt kann durch Sinnesorgane oder technische Instrumente niemals vollständig wahrgenommen werden. Speziell im Bereich der Bilder kann durch das menschliche Auge bzw. durch Kamerasensoren nur ein eingeschränktes Spektrum an elektromagnetischen Wellen erfasst werden. Somit wird die Umwelt auf einen Problemkreis $\Omega \subset U$ beschränkt, dessen Muster ${}^{\rho}\mathbf{f}(\mathbf{x})$ mit geeigneten Sensoren wahrgenommen werden. Formal kann der Problemkreis Ω definiert werden als

$$\Omega = \{{}^{\rho}\mathbf{f}(\mathbf{x}) | \rho = 1, 2, \dots\} \subset U. \quad (2.2)$$

[Nie03] erweitert diese Definition durch die Verwendung von mehreren Sensoren und den zugehörigen Parametern für Aufnahmebedingungen. Da in der vorliegenden Arbeit nur bereits erfasste Bilder verwendet werden und somit kein Einfluss auf die Anzahl

der Sensoren oder die Aufnahmebedingungen genommen werden kann, werden diese Erweiterungen für die Definition des Problemkreises Ω nicht weiter ausgeführt.

Muster

Ein Muster $\mathbf{f}(\mathbf{x})$ lässt sich als Funktion

$$\mathbf{f}(\mathbf{x}) = \begin{pmatrix} f_1(x_1, \dots, x_n) \\ f_2(x_1, \dots, x_n) \\ \vdots \\ f_m(x_1, \dots, x_n) \end{pmatrix} \quad (2.3)$$

oder äquivalent als eine Menge von Tupeln

$$\mathbf{f} = \{(\mathbf{x}, \mathbf{f})^\top \mid \forall \mathbf{x}, \forall \mathbf{f}\} = \{(\mathbf{x}, \mathbf{f}_\mathbf{x})^\top\} \quad (2.4)$$

auffassen. Im Fall von digital aufgenommenen oder digitalisierten Bildern liegen \mathbf{f} und \mathbf{x} bereits im diskreten Wertebereich vor und bilden somit die Rohdaten. Die Quantisierung von kontinuierlichen Signalen wird in der vorliegenden Arbeit somit nicht betrachtet. Ein Farbfoto im RGB-Farbraum kann z. B. durch die Funktionen $f_r(x,y)$, $f_g(x,y)$ und $f_b(x,y)$ charakterisiert werden ($m = 3$, $n = 2$), wobei die Funktionen den Rot-, Grün- und Blauwert für ein gegebenes Pixel (x,y) bestimmen. Schwarz-Weiß-Fotos können als ein Muster mit einer einzigen Funktion $f(x,y)$ aufgefasst werden ($m = 1$, $n = 2$), wobei die Funktion den Grauwert für das Pixel (x,y) angibt.

Klasse

Bevor auf die Definition der Klassifikation eingegangen wird, wird zuerst der Begriff der Klasse eingeführt. Klassen Ω_κ ergeben sich durch eine Zerlegung des Problemkreises $\Omega \subset U$ in k – bzw. bei Verwendung einer Rückweisungsklasse Ω_0 in $k+1$ – Teilmengen, wobei folgende Eigenschaften erfüllt sein müssen:

$$\begin{aligned} \Omega_\kappa &\neq \emptyset \quad \kappa = 1, \dots, k, \\ \Omega_\kappa \cap \Omega_\lambda &= \emptyset \quad \kappa \neq \lambda, \\ \text{und } \bigcup_{\kappa=1}^k \Omega_\kappa &= \Omega \quad \text{bzw. } \bigcup_{\kappa=0}^k \Omega_\kappa = \Omega. \end{aligned} \quad (2.5)$$

Im Fall der Objekterkennung ist Ω die Vereinigungsmenge der dem System zum Zeitpunkt t bekannten Objektklassen Ω_κ . Gemäß den Bedingungen in Gleichung 2.5 enthalten die einzelnen Objektklassen Ω_κ somit eine disjunkte Teilmenge der Muster ${}^{\rho}\mathbf{f}(\mathbf{x})$ des Problemkreises Ω . Je nach Problemdefinition können auch Mischklassen eingeführt werden oder die Menge Ω hierarchisch über mehrere Stufen zerlegt werden.

Klassifikation

Ziel der Klassifikation ist es basierend auf der Beobachtung (Menge von Mustern ${}^{\rho}\mathbf{f}(\mathbf{x})$) die zugehörige Klasse Ω_κ aus k möglichen Klassen zu bestimmen. Je nach Komplexität des Musters bzw. Beschaffenheit der Klassifikationsaufgabe kann unter folgenden Fällen unterschieden werden:

$$\mathbf{f} \Rightarrow \begin{cases} \Omega_\kappa & \text{eine Klasse,} \\ \boldsymbol{\Omega} = [{}^1\Omega, \dots, {}^N\Omega] & \text{eine Folge von Klassen,} \\ (\Omega_\kappa, \mathbf{T}_\kappa, \mathbf{R}_\kappa) & \text{Klasse und Lokalisation,} \end{cases} \quad (2.6)$$

wobei \mathbf{T} die Translation und \mathbf{R} die Rotation des Referenzpunktes eines Musters bzgl. eines Referenzkoordinatensystems angibt. Innerhalb der Objekterkennung werden ähnlich zu der Unterscheidung in Gleichung 2.6 die Probleme der Lokalisierung und der Identifikation von Objekten in Bildern getrennt.

Analyse

Zusätzlich zur Klassifikation kann auch eine tiefergehende Analyse der Muster ${}^{\rho}\mathbf{f}(\mathbf{x})$ erfolgen, bei dem den Mustern symbolische Beschreibungen ${}^{\rho}\mathcal{B}$ zugeordnet werden. Die Beschreibung besteht dabei aus einem Netzwerk oder einer formalen Datenstruktur, welche die aus dem Muster ${}^{\rho}\mathbf{f}(\mathbf{x})$ berechneten Instanzen I_j von Konzepten C_k eines Modells oder einer Wissensbasis enthält, formal also ${}^{\rho}\mathcal{B} = \langle I_j(C_k) \rangle$. Je nach verfolgtem Ziel der Analyse kann die Beschreibung ${}^{\rho}\mathcal{B}$ von einer diagnostischen Interpretation über eine automatische Reaktion bis zu einer Liste von Objekten oder Ereignissen variieren. Letztere sind insbesondere für die Annotation von Bildern relevant. Textuelle Beschreibungen von Bildern werden in Kapitel 3 ausführlich untersucht und die Anforderungen an die Annotation in Abschnitt 3.2.3 und 3.4 zusammengefasst.

Merkmal

Ein Muster ${}^{\rho}\mathbf{f}(\mathbf{x})$ wird durch einen Merkmalsvektor ${}^{\rho}\mathbf{c}$ beschrieben, welcher für seine Zugehörigkeit zu einer Klasse Ω_{κ} charakteristisch ist. Merkmale sind somit möglichst kompakte Beschreibungen von Mustern mit dem Ziel die zugeordneten Klassen gut darzustellen. Dabei können Merkmale von relativ einfachen Beschreibungen eines Musters (z. B. Höhe und Breite eines Bildes) bis zu komplexen Merkmalen (z. B. Beschreibung einer Menge von Bildbereichen durch Histogramme von Gradienten) variieren. Da der Merkmalsvektor ${}^{\rho}\mathbf{c}$ direkt aus dem Muster ${}^{\rho}\mathbf{f}(\mathbf{x})$ ermittelt wird, kann die Klassifikation des Musters ${}^{\rho}\mathbf{f}(\mathbf{x})$ auch als eine Abbildung ${}^{\rho}\mathbf{c} \rightarrow \kappa \in \{1, \dots, k\}$ (bzw. ${}^{\rho}\mathbf{c} \rightarrow \kappa \in \{0, 1, \dots, k\}$) bei der Verwendung einer Rückweisungsklasse Ω_0 formuliert werden. Merkmale sollten gemäß der Kompaktheithypothese ausgewählt werden. Sie besagt, dass die Merkmale der dem Muster \mathbf{f} zugeordneten Klasse Ω_{κ} einen möglichst kompakten Bereich im Merkmalsraum einnehmen und dabei die von verschiedenen Klassen eingenommenen Bereiche einigermaßen getrennt sein sollten.

Muster werden als ähnlich gesehen, wenn ihre Merkmale sich nur wenig unterscheiden, d. h. die Abstände bleiben unterhalb einer vorgegebenen Schwelle. Basierend auf diesem Ähnlichkeitskriterium können Klassen automatisch gebildet und neue Muster bekannten Klassen zugeordnet werden. Wichtig dabei ist die Ähnlichkeit, d. h. die Metriken, die Distanzmaße und die Gewichtung so zu wählen, dass dabei die für den Anwender relevanten Ähnlichkeiten widergespiegelt werden.

Bei Bildern bestehen die Rohdaten bzw. die Muster ${}^{\rho}\mathbf{f}(\mathbf{x})$ aus einer großen Pixel-Matrix, aus der zur Beschreibung der Bilder Merkmale (Features) ${}^{\rho}\mathbf{c}$ extrahiert werden. Die Merkmale ergänzen somit die Beschreibungsdaten der Bilder. Merkmale sind z. B. Farbhistogramme, Texturen, Formen oder verschiedene Beschreibungen interessanter Punkte im Bild. Die Merkmalsextraktion kann sich dabei entweder auf das gesamte Bild beziehen oder nur auf einen Teilbereich. Zur Bestimmung des Teilbereichs kann eine feste Aufteilung (Grid) für das Bild verwendet werden. Eine andere Möglichkeit ist das Bild manuell, semi- oder vollautomatisch zu segmentieren, womit eine genauere Einschränkung auf die Objekte im Bild ermöglicht wird. Wichtig bei den extrahierten Merkmalen ist, dass sie möglichst kompakt und unterscheidungskräftig sind. Außerdem sollen sie möglichst tolerant bzgl. Farbverschiebungen und unabhängig von verschiedenen Belichtungen, Ausrichtungen und Perspektiven die Bilder bzw. Teilbilder (hier Objekte) repräsentieren.

Stichprobe

Um Klassen erlernen zu können wird eine Menge von Beobachtungen des Problemkreises Ω mit Zusatzinformationen benötigt. Diese wird Stichprobe oder Trainingsmenge genannt und kann wie folgt definiert werden:

$$\omega = \{({}^1\mathbf{f}(\mathbf{x}), y_1), \dots, ({}^N\mathbf{f}(\mathbf{x}), y_N)\} \subset \Omega, \quad (2.7)$$

wobei ${}^\rho\mathbf{f}(\mathbf{x})$ für das ρ -te Muster und y_ρ für die zugehörige Zusatzinformation steht. In der Objekterkennung besteht die Zusatzinformation zu einem gegebenem Bild ${}^\rho\mathbf{f}(\mathbf{x})$ bei den meisten Datensätzen aus der zugehörigen Klasse, also $y_\rho \in \{1, \dots, \kappa, \dots, k\}$. Manche Datensätze geben zu einem Bild auch mehrere Klassen (bzw. frei gewählte Begriffe, die nicht durch eine Taxonomie eindeutig zu Konzepten zugeordnet sind) an, zum Teil nur als Liste $(y_\rho : \{\Omega_{\rho,1}, \dots, \Omega_{\rho,k_\rho}\} \in {}^\rho\mathbf{f}(\mathbf{x}))$, zum Teil auch mit zusätzlicher räumlicher Information $(y_\rho : \{(\Omega_{\rho,1}, \mathbf{T}_{\rho,1}), \dots, (\Omega_{\rho,k_\rho}, \mathbf{T}_{\rho,k_\rho})\} \in {}^\rho\mathbf{f}(\mathbf{x}))$.

Basierend auf den Beobachtungen der Stichprobe kann durch verschiedene Klassifikationsmethoden die Abbildung ${}^\rho\mathbf{f}(\mathbf{x}) \Rightarrow \Omega_\kappa \in \Omega \subset U$ optimiert werden. Dabei gilt es diejenige Entscheidungsregel δ – d. h. diejenigen Merkmale und Klassifikatoren und deren Parameter – zu finden, welche die mittleren Kosten V minimiert, also $\delta^* = \arg \min_\delta V(\delta)$. Bei der Objekterkennung wird in der Regel versucht die Auswahl der Merkmale bzw. der Klassifikatoren so zu bestimmen, dass dabei verschiedene Bewertungsmaße maximiert bzw. minimiert werden. Übliche Bewertungsmaße werden in Abschnitt 4.1.4 vorgestellt. In dieser Arbeit werden zur Optimierung des vorgestellten Ansatzes bzw. für die Vergleiche zu anderen Verfahren vorwiegend die Bewertungsmaße Area Under the ROC Curve (AUC), Average Precision (AP), Mean Average Precision (MAP) und der hierarchische Fehler mittels WordNet eingesetzt.

Abbildung 2.1 zeigt den Ablauf der Klassifikation von Mustern basierend auf den bislang eingeführten Begriffen.

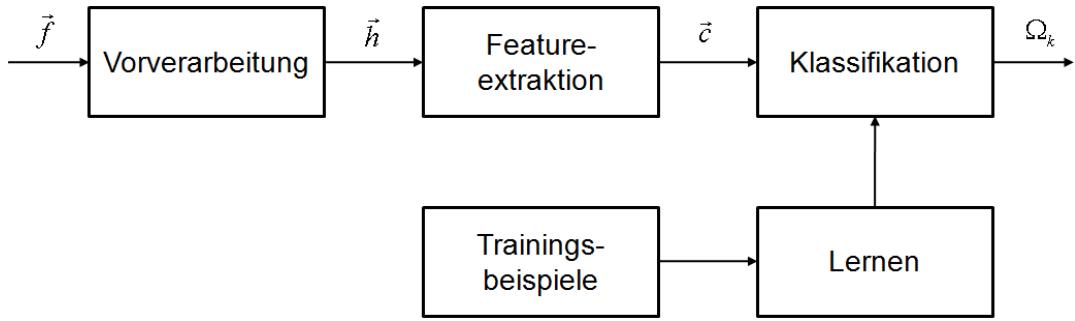


Bild 2.1: Grundsätzliches Vorgehen bei Klassifikationsproblemen nach [Nie03].

2.3 Problemdefinition

In diesem Abschnitt werden basierend auf den Eigenschaften und Begriffsdefinitionen der vorherigen Abschnitte die in der vorliegenden Arbeit behandelten Probleme abgegrenzt und genau spezifiziert.

Erweiterbarkeit

Ausgehend von der Definition für den Begriff der Erweiterbarkeit in Abschnitt 2.1.1 muss ein Klassifikationssystem, welches kontinuierlich und dynamisch wachsen soll, in der Lage sein neue Klassen einfach und schnell integrieren zu können. Im Wesentlichen heißt das, dass der Problemkreis Ω um eine neue Klasse $\Omega_\nu \notin \Omega$ erweitert wird und somit ein neuer Problemkreis $\Omega' = \Omega \cup \{\Omega_\nu\}$ entsteht. Bei diesem Erweiterungsschritt werden keine bereits bestehenden Klassen aufgelöst. Es können lediglich Klassen auf gleicher, übergeordneter oder untergeordneter Ebene hinzugefügt werden. Beispielsweise kann zur Klasse Obst als Unterklasse Apfel und Orange neu hinzugefügt werden, die Klasse Obst bleibt jedoch weiterhin bestehen. Die Erweiterung um neue Klassen ist ein signifikanter Eingriff, da viele Klassifikationsmethoden von einer statischen Menge Ω ausgehen und bei einer Änderung von Ω vollständig neu aufgesetzt werden müssen.

Weiterhin ist die Erweiterbarkeit bereits bei der Beschreibung von Mustern ${}^p\mathbf{f}$ durch Merkmale ${}^p\mathbf{c}$ im Auge zu behalten. Einige Merkmale werden in direktem Bezug auf den eingeschränkten Problemkreis Ω extrahiert. Bei einer statischen Menge Ω kann diese Vorgehensweise zu Verbesserungen bei der Klassifikation führen. In einem stetig dynamisch erweiterbaren Problemkreis müssen diese Merkmale jedoch bei jeder Änderung von Ω neu berechnet werden, was zu einem beträchtlichen Mehraufwand führen kann. Ein Beispiel für das Problem der Erweiterbarkeit in der Objekterkennung ist das visuelle Vokabular,

auf dem die Histogramme von visuellen Wörtern und letztendlich die Klassifikatoren aufbauen. Die erwähnten Merkmale werden in Abschnitt 4.1.2 bzw. das Problem der Erweiterbarkeit bzgl. dieser Merkmale in Abschnitt 5.1.1 ausführlich vorgestellt und diskutiert. Erstrebenswert ist deswegen eine Beschreibung der Muster (Bilder), welche sich entweder direkt ausschließlich auf die Klasse Ω_κ bezieht oder unabhängig vom Problemkreis Ω ist, gleichzeitig jedoch wenig Einbußen bei der Klassifikation mit sich bringt.

Skalierbarkeit

Der Begriff der Skalierbarkeit wurde in Abschnitt 2.1.2 definiert als die Fähigkeit des Systems bzgl. einer oder mehrerer Dimensionen „skalieren“ zu können. Für die Klassifikation selbst kann die Skalierbarkeit sowohl als eine sehr große Stichprobe ω oder als eine hohe Anzahl k von Klassen Ω_κ innerhalb des Problemkreises Ω aufgefasst werden.

In der Objekterkennung wird unter der Skalierbarkeit eines Systems meistens die Verwaltung von einer hohen Anzahl von Bildern oder möglichst vielen Klassen verstanden. Gleichzeitig wird eine möglichst gute Erkennungsrate angestrebt, wobei die Erweiterbarkeit und die Antwortzeit in der Regel vernachlässigt werden. Auch gibt es zur Zeit leider keinen Konsens darüber, was genau als „hohe Anzahl“ anzusehen ist. Die Anzahl k der Klassen variiert für gewöhnlich zwischen 100 und 10 000.

Ähnlich zum Problem der Erweiterbarkeit kann auch für die Skalierbarkeit eines Systems sowohl bei der Klassifikationsmethode als auch bei der Auswahl der Merkmale angesetzt werden. Merkmale, welche eine kleine Anzahl von Klassen im Merkmalsraum gut von einander trennen, können bei einer hohen Anzahl von Klassen wegen einer dichten Besiedlung des Merkmalsraums ggf. die Klassen weder kompakt darstellen noch sauber abgrenzen. Für Klassifikationsmethoden wurde in [DBLFF10] gezeigt, dass ein Vorgehen, welches auf kleinen Datensätzen andere Ansätze bzgl. der Erkennungsrate (deutlich) übertrifft, bei einer hohen Anzahl von Klassen schlechter abschneidet als die gleichen Ansätze.

In dieser Arbeit wird neben der Erweiterbarkeit des Systems auch eine möglichst gute Erkennungsrate (gemessen durch AUC und MAP) und damit verbunden eine möglichst genaue Annotation von Bildern bei einer hohen Anzahl von Klassen angestrebt. Die Skalierbarkeit des Verfahrens wird auf Datensätzen unterschiedlicher Größe evaluiert. Für die Skalierung auf eine hohe Anzahl von Klassen wird das Verfahren auf einem Datensatz mit mehr als 3 000 Klassen evaluiert.

2.4 Zusammenfassung

In diesem Kapitel wurde die Terminologie für die vorliegende Arbeit festgelegt. Aufbauend auf den Definitionen wurden die zu lösenden Probleme genau spezifiziert und abgegrenzt.

KAPITEL 3

ANNOTATION VON BILDERN

Neben dem Problem der effizienten Verwaltung von großen Bilddatenmengen stellt sich die Frage nach wirkungsvollen Suchmöglichkeiten. Um letzteres zu erreichen, müssen die Bilder indiziert bzw. annotiert werden, was jedoch stark von der Suche abhängig ist. Unter Annotation von Bildern wird die textuelle Beschreibung des Bildinhalts mittels Schlüsselwörtern oder eines Fließtextes verstanden. Die für die textbasierte Suche benötigten Beschreibungen können durch manuelle Annotation, durch Anwendung eines Katalogs oder einer Ontologie, semi-automatisch durch Unterstützung von speziellen Algorithmen oder vollautomatisch mittels einer Wissensdatenbank geschehen. Sofern die Anfragen eher in Form von Bildern, Grafiken oder Merkmalen gestellt werden, ist eine inhaltsbasierte Indizierung unumgänglich. Dies geschieht meistens durch die Beschreibung der Bilder durch ein Modell bzw. durch geeignete Merkmale.

Mehrere Anwendungsfelder profitieren von Bildannotationen. Im Bericht [IAS] des World Wide Web Consortium (W3C) zur semantischen Annotation von Bildern im Web werden fünf Beispielszenarien genannt: private Bildsammlungen, das digitalisierte kulturelle Erbe, Fernseharchive, biomedizinische Bilder und umfangreiche Bildsammlungen der NASA (Messbilder). Die größten Triebkräfte zur Annotation von Bildern sind die textbasierte Suche nach multimedialen Inhalten und die Darstellung von Webseiten mit Bildern für Menschen mit Sehbehinderungen. Diese werden in Abschnitt 3.1.1 und Abschnitt 3.1.2 näher diskutiert.

Die zu einem Bild annotierten Attribute können auf verschiedenste Weise kategorisiert werden. Auch die Relevanz einzelner Attributgruppen ist bei der Suche bzw. bei der Beschreibung von Bildern unterschiedlich. Diese Themenbereiche werden in Abschnitt 3.2.1 und Abschnitt 3.2.2 untersucht. Anschließend wird ein Anforderungskatalog für die textuelle Annotation von Bildern ermittelt.

Speziell für die Verarbeitung von Bildern, die mittels der digitalen Fotografie entstanden sind, haben sich in den letzten Jahren drei Metadatenstandards etabliert: Exchangeable image file format (Exif) (bzw. Tagged Image File Format (TIFF) 6.0), IPTC Information Interchange Model (IPTC-IIM) und Extensible Metadata Platform (XMP). Außerdem wurden zur einheitlichen Annotation verschiedene Konzept-Thesauri erstellt. Diese werden in Abschnitt 3.3 kurz vorgestellt.

Abschließend werden in Abschnitt 3.4 die Erkenntnisse aus der textuellen Annotation von Bildern zusammengefasst.

3.1 Motivation für Textannotationen

Ein bekanntes Sprichwort besagt, dass Bilder mehr als tausend Worte darstellen können. Auch ist es aus zahlreichen Beispielen bekannt, dass ein Bild die Information z. B. zu einer Tatsache, Funktionsweise oder Struktur häufig viel besser und intuitiver wiedergeben kann als die textuelle Formulierung. Genau aus diesen Gründen ist es auch schwierig eine möglichst vollständige und präzise Beschreibung zu Bildern zu finden. Die textuelle Annotation von Bildern wird jedoch einerseits zum Auffinden der Bilder zu einem späteren Zeitpunkt durch die textbasierte Suche sowie zur Beschreibung von Bildinhalten für Menschen mit Sehbehinderungen benötigt. Im Folgenden werden diese zwei wesentlichen Motivationsfaktoren näher erläutert.

3.1.1 Textbasierte Suche auf Bildern

Im Bildretrieval können Anfragen auf mehreren Arten gestellt werden. Während [BVBF07] die Suche nur aus der Benutzersicht betrachtet, werden in [DJLW08] die Suchmöglichkeiten sowohl aus der Benutzer- als auch aus der Systemperspektive beschrieben. Somit ergeben sich aus der Benutzersicht die Möglichkeiten der Suche mittels Schlüsselwörtern, Freitext, Bildern, Grafiken und der Kombination der genannten Anfragetypen. Aus der Systemperspektive kann die text- und die inhaltsbasierte Suche

unterschieden werden, die auch kombinierbar sind. Weiterhin ist die Rückmeldemöglichkeit des Benutzers bzgl. des Anfrageergebnisses (Relevance Feedback) zu erwähnen, die von einem System, welches Interaktivität unterstützt, implementiert werden muss. Das Relevance Feedback kann sich dabei auf nur eine Modalität (Text oder visuelle Information) oder auf die Kombination mehrerer Modalitäten beziehen.

In [VWS01] wurde eine andere Klassifikation der Anfragemöglichkeiten im Bildretrieval erstellt. Anfragen können demnach grob drei Kategorien zugeordnet werden, die weiter unterteilbar sind:

- Anfragen mittels Abstraktion: Der Benutzer gibt konkrete Fakten (z. B. Schlüsselwörter, Farbwerte) in einem vordefinierten Format ein. Diese Anfragen können weiter unterteilt werden auf Anfragen mittels symbolischer Beschreibungen und Anfragen mittels Bildmerkmalen. Ersteres bezieht sich auf informationstragende Attribute, die sich aus dem Kontext oder der Interpretation des Bildes ergeben (z. B. textuelle Annotation). Letzteres sind lediglich Daten bzw. Werte für Merkmale, welche direkt aus den Bilddaten ermittelt wurden (z. B. Farbwerte).
- Anfragen mittels eines Beispielbilds: Der Benutzer gibt als Anfrage ein Beispielbild an. Je nachdem woher das Beispielbild stammt bzw. wie es entstanden ist, können Anfragen mittels eines externen Beispielbildes, Anfragen mittels eines systeminternen Beispielbildes und Anfragen mittels konstruierter Bilder (z. B. Skizze, Umriss, ggf. mit Farbe kombiniert) unterschieden werden.
- Anfragen mittels Navigation: Im wesentlichen handelt es sich hierbei um das Browsen (Umsehen, Schmökern) nach Bildern in einer Bilddatenbank. Je nachdem inwieweit der Inhalt bzw. die Metadaten oder sonstige Verwaltungsstrukturen zum Auffinden von ähnlichen Bildern berücksichtigt werden, kann nach Anfragen mittels Assoziation und visueller Untersuchung unterschieden werden.

Die textbasierte Suche kann als Anfrage mittels symbolischer Beschreibungen aufgefasst werden, da hierbei nach Darstellungen gesucht wird, die den Bildinhalt textuell interpretieren bzw. beschreiben.

Der Vorteil der textbasierten Suche ist, dass bei Vorhandensein von Schlüsselwörtern zu Bildern die Anfragen dank jahrzehntelanger Forschung im Bereich von Datenbanken und Text Retrieval rasch beantwortet werden können. Während die Indizierung von Text bereits ausgiebig untersucht und auch in diversen Suchmaschinen wie z. B. Google und Yahoo! erfolgreich etabliert wurde, blieb nach [DJLW08] die umfangreiche inhalts-

basierte Indizierung von Bildern von den zahlreichen Veröffentlichungen der visuellen Beschreibungen von Bildern überschattet. Wegen der stetig wachsenden Zahl an digitalen Inhalten wird der Anspruch auf effiziente Indizierung von Bildern immer größer, da diese entscheidend für Systeme mit großen Datenbeständen sind. Eine nähere Untersuchung der aktuellen Forschungsergebnisse auf dem Gebiet der textbasierten Suche und der effizienten Indizierung von Bildern erfolgt in Abschnitt 3.2.

Für textbasierte Anfragen sind zuverlässige Metadaten vonnöten, die manuell, semi- oder vollautomatisch ermittelt werden müssen. Heutige Suchmaschinen bieten ausschließlich eine textbasierte Suche nach Bildern an, die jedoch sehr stark von den manuell zugeordneten HTML-Tags oder vom Kontext des Bildes abhängt. Dies hat den Grund, dass beim Letzteren nur die von der allgemeinen Websuche bekannten und etablierten automatischen Textindizierungsverfahren zum Einsatz kommen. Zwar können dadurch Beschreibungen zum Bild ermittelt werden, jedoch kann der Text auch teilweise oder völlig unabhängig vom Bild sein, oder von Themen handeln, die nur indirekt mit dem Bild zusammenhängen, wodurch die aus dem Kontext ermittelte Annotation des Bildes nicht zutreffend sein kann. Auch die im Kontext verwendeten Begriffe und Schlüsselwörter können stark variieren, da sie individuell von Menschen erstellt werden, womit die Suche nur einen Bruchteil der potenziellen Ergebnismenge zurückliefern kann. Zuletzt gibt es z. B. bei privaten Fotosammlungen auch das Problem, dass gar keine Kontextdaten verfügbar sind, welche die Bilder beschreiben würden, man jedoch trotzdem nach Bildern in diesen Archiven suchen möchte. Zur Zeit erfolgt bei großen Bildsuchmaschinen wie Google oder Flickr noch keine Indizierung nach den eigentlichen Bildinhalten. Wünschenswert wäre die automatische Generierung einer zuverlässigen Annotation basierend auf dem Bildinhalt.

3.1.2 Bilder für Menschen mit Sehbehinderungen

Nach einer Studie der World Health Organization (WHO) [RPE⁺04] gab es im Jahr 2002 161 Millionen Menschen mit Sehbehinderungen, von denen 37 Millionen vollständig blind waren. Für Deutschland belaufen sich die Schätzungen nach [Ber05] auf insgesamt 1,2 Millionen Sehbehinderte. Blinde und sehbehinderte Menschen müssen im Alltag mit mehreren besonderen Schwierigkeiten zurecht kommen. Rechnergestützte Verfahren können nach [MC12] vor allem bei folgenden Gebieten eine Hilfestellung bieten:

- *Mobilität*: sichere und komfortable Bewegung in der Umgebung.
- *Orientierung*: Bewusstsein über die aktuelle Position innerhalb der Umgebung sowie Wegfindung zum Ziel.
- *Zugriff auf schriftliche Information*: Lesen von Druckerzeugnissen, Schildern, Beschriftungen (z. B. auf Lebensmitteln), (Warn-)Hinweisen sowie Anzeigen (z. B. LED- oder LCD-Anzeigen auf elektronischen Geräten).

Wichtig bei der Unterstützung ist, dass die Person nicht mit unnützen Informationen überschwemmt wird, die Geräte möglichst unauffällig und relativ kostengünstig sind sowie die Genauigkeit verlässlich genug ist. Im Gegensatz zu Menschen ohne Sehbehinderungen kommen zusätzliche Schwierigkeiten durch das Sichtfeld der verwendeten Kamera hinzu. Blinde und Sehbehinderte können nur bedingt oder überhaupt nicht mit der Kamera des unterstützenden Gerätes auf relevante Objekte oder Texte zielen, was als zusätzliche Schwierigkeit bei der rechnergestützten Erkennung von Texten und Objekten berücksichtigt werden muss.

Zahlreiche mehr oder weniger umfangreiche Untersuchungen und Studien auf dem Gebiet der Bildbetrachtung für Menschen mit Sehbehinderungen belegen, dass multimediale Webinhalte mit eingeschränktem Sehvermögen noch bei weitem nicht hinreichend barrierefrei zugänglich sind. Diese Webnutzer lassen die betrachteten Webseiten durch sog. Screen Reader Werkzeuge (z. B. JAWS¹) vorlesen. Bei Bildern werden die `alt` und `longdesc` HTML-Tags vorgelesen, welche alternative textuelle Beschreibungen für multimediale Objekte beinhalten – bzw. beinhalten sollten. Sowohl in der Version 1.0 als auch in der Version 2.0 der Web Content Accessibility Guidelines (WCAG) [WCA] des W3C wird die Angabe von sinnvollen textuellen Beschreibungen für nicht-Textelemente an allererster Stelle genannt, was die Wichtigkeit von Bildannotationen hervorhebt.

Ausgehend von den Anforderungen des WCAG wurde in mehreren Studien nachgeprüft, inwieweit die `alt`-Tags von Bildern mit Beschreibungen gefüllt wurden. In den meisten Untersuchungen werden die Bilder auf informative und dekorative Gruppen aufgeteilt. In [PHD05] wurden jeweils 10 Webseiten aus 10 verschiedenen Bereichen untersucht und anschließend in Interviews mit 5 blinden Webnutzern aus unterschiedlichen Altersgruppen ausgewertet. Rund 70,8% der informativen Bilder hatten eine Beschreibung, die jedoch in den meisten Fällen sehr dürftig ausfiel. Zwar wurde in dieser Arbeit nur eine kleine

¹ <http://www.freedomscientific.com/products/fs/jaws-product-page.asp>

3. Annotation von Bildern

Untersuchung durchgeführt, jedoch sind die Erkenntnisse aus den Interviews mit blinden Webnutzern sehr wertvoll, da hierbei eine Art Anforderungskatalog ermittelt werden konnte.

Nach Aussagen der blinden Personen sollten nicht alle Bilder einer Webseite beschrieben werden. Ausnahmen bilden z. B. dekorative Bilder, Platzhalter, Aufzählungszeichen, Logos und Bilder, die im Text ausführlich beschrieben werden. Der gewünschte Umfang der Beschreibung von informativen Bildern ist zwar kontextabhängig, nach Aussagen der befragten Personen sollten jedoch folgende Elemente auf jeden Fall erwähnt werden, wobei die Reihenfolge der Wichtigkeit für die befragten blinden Personen entspricht:

- Objekte, Gebäude, Personen im Bild,
- Handlung, Ereignisse im Bild,
- Ziel des Bildes,
- Farben im Bild,
- Gefühle verbunden mit dem Bild und Atmosphäre des Bildes,
- der abgebildete Ort.

Dabei sollte die wichtigste Information in einfacher Sprache direkt am Anfang stehen. Die Länge der Beschreibungen war für die Benutzer uninteressant, solange ein gutes Gleichgewicht zwischen Qualität und Quantität angestrebt wurde. Zu lange Beschreibungen wurden nicht vollständig durchgelesen, wobei Beschreibungen aus 2-3 Wörtern als zu kurz empfunden wurden. Ein Einverständnis herrschte darüber kürzere Beschreibungen im `alt`-Tag von Bildern durch `longdesc`-Tags zu ergänzen, welche bei Bedarf mit vorgelesen werden können. Als generelles Problem wurde die bisherige Berücksichtigung der WCAG genannt, da die meisten Mainstream-Webseiten keine oder nur sehr schlechte Beschreibungen enthielten.

In einer anderen Studie von [BKL⁺⁰⁶] wurde die Auswirkung der Popularität einer Webseite auf die Beschreibungen der Bilder ausgewertet. Die Aufteilung der Bilder erfolgte ähnlich wie bei [PHD05] auf informative und dekorative Bilder, jedoch wurden hier die beiden Gruppen als signifikant und nicht signifikant benannt. Insgesamt wurden 49 665 Bilder auf 945 Webseiten untersucht, wobei auf den Top 500 Webseiten lediglich 39,6% der signifikanten Bilder eine wirklich nutzbare Beschreibung enthielten. Auf den jeweiligen Webseiten der einzelnen US Bundesstaaten betrug diese Zahl 82,5%. In einer weiteren Studie wurde der Webverkehr des eigenen Departments eine Woche lang überwacht, wobei 4,9 Millionen signifikante Bilder erfasst wurden. 63,2% der Bilder hatten eine

Beschreibung. Auf die Qualität dieser Beschreibungen wird in [BKL⁺06] leider nicht weitergehend eingegangen.

In einer allgemeiner angelegten Untersuchung verglichen [BCB⁺07] das Verhalten von blinden und normalen Webnutzern über einen Zeitraum von 7 Wochen. Insgesamt wurden 337 036 Bilder abgerufen, 109 624 davon von Blinden. Lediglich 56,9% der Bilder enthielten eine Beschreibung. Signifikant ist der Unterschied bei klickbaren Bildern: 72,17% der Bilder, auf die die blinden Versuchspersonen geklickt haben, enthielten eine Beschreibung, während diese Zahl bei den normalen Benutzern nur 34,03% betrug. Blinde interagieren demnach nachvollziehbarer Weise viel weniger mit Webinhalten, die für sie nur schwer zugänglich sind.

Es existieren noch weitere Studien, die jedoch keine neuen Erkenntnisse zu den bisherigen beitragen. In [Cra06] wurden verschiedene Feinheiten bzgl. der Verwendung von `alt`-Tags zusammengestellt. Ferner sind hier auch einige Statistiken zu Bildern und `alt`-Tags in Webseiten zu finden. In [KLB01] wurden Webbilder, die hauptsächlich Text enthalten, untersucht. [ST09] deutet auf zahlreiche Einschränkungen bzgl. dem Umgang mit PCs bei blinden Personen hin – unter anderem auch auf die Schwierigkeit Bilder korrekt zu interpretieren.

3.1.3 Zusammenfassung

Die fehlende inhaltsbasierte Indizierung von Bildern in großen Suchmaschinen, die am meisten verbreitete textbasierte Suchmöglichkeit nach Bildern und die zahlreichen Studien bzgl. dem Browsing-Verhalten von Benutzern mit eingeschränktem Sehvermögen belegen eindeutig den enormen Bedarf von automatisch erstellten Bildannotationen und der effizienten, skalierenden Indizierung von großen Bildsammlungen.

3.2 Textbasierte Suche auf Bildern

Die textuelle Annotation von Bildern dient primär zum Auffinden der Bilder zu einem späteren Zeitpunkt durch die textbasierte Suche. Um die Annotation möglichst präzise auf dieses Ziel auszurichten werden in diesem Abschnitt basierend auf den Erkenntnissen verschiedener Untersuchungen zur textbasierten Bildsuche die Anforderungen an die Bildannotationen ermittelt. Zuerst werden in Abschnitt 3.2.1 verschiedene Untersuchungen in der Literatur zur textbasierten Suche vorgestellt. Anschließend werden

in Abschnitt 3.2.2 die möglichen Beschreibungen und Attribute von Bildern logisch nach ihren Eigenschaften klassifiziert. Basierend auf diesen Ergebnissen werden die Anforderungen an die Annotation von Bildern in Abschnitt 3.2.3 zusammengefasst.

3.2.1 Studien zu textuellen Benutzeranfragen auf Bilddatenbanken

In diesem Abschnitt werden verschiedene Studien auf diversen, mehr oder weniger speziellen Bilddatenbanken vorgestellt. Es wird versucht zu ermitteln, wie die textbasierte Suche nach Bildern charakterisiert werden kann, d. h. wie und wonach die Benutzer suchen bzw. welche Merkmale am meisten zum textbasierten Suchvorgang beitragen. Grundsätzlich lassen sich die Studien nach der Art der Bilddatenbanken unterteilen. Die meisten Untersuchungen wurden auf Bilddatenbanken für die Kunstgeschichte und für den Journalismus durchgeführt. Erst in letzter Zeit kamen Auswertungen zur textbasierten Bildsuche im Internet hinzu.

Eine der ersten Studien zur textbasierten Suche in Bilddatenbanken wurde in [Ens93] durchgeführt. Mehr als 2 700 Anfragen an die Hulton Deutsch Collection Limited Datenbank (Bilddatenbank für Journalismus) wurden analysiert zur Ermittlung:

- ob eine Anfrage zum Auffinden einer Person, eines Objekts oder eines Ereignisses diente und
- ob eine Anfrage bzgl. der Aspekte Zeit, Ort, Aktion oder technischen Angaben weiter verfeinert wurde.

Das vorgegebene Klassifikationsschema der Datenbank begünstigte die Suche nach Personen, Objekten und Ereignissen. 70% der Anfragen waren dementsprechend auch auf bestimmte Personen, Objekte und Ereignisse ausgerichtet. Sofern eine Verfeinerung nötig war, wurde dies in den meisten Fällen durch die Angabe der Zeit bewerkstelligt.

[Orn95] hat Anfragen auf 13 dänischen Zeitungsbildarchiven mit 25 Archivaren und 26 Journalisten untersucht. Die Analyse der Benutzeranfragen ergab, dass in 50% der Fälle nach bekannten Personen gesucht wurde. Die restlichen Anfragen bezogen sich auf Hintergrundinformationen (Ort, Zeitpunkt), spezielle Ereignisse, Emotionen, bekannte Gebäude und Länder. Je nach Anfrageart wurden 5 Benutzergruppen unterschieden: spezielle, generische, geschichtenerzählende, geschichtengebende und lückenfüllende Personen. Eine detaillierte Beschreibung dieser Journalismus-spezifischen Gruppen wird hier nicht weiter ausgeführt und ist in [Orn95] zu finden.

In der Studie von [AE97] wurden mehr als 1 700 Anfragen auf 7 verschiedenen Bilddatenbanken untersucht und nach den in Abschnitt 3.2.2 beschriebenen zwölf Kategorien eingeordnet. Die Untersuchungen ergaben, dass die meisten Anfragen sich auf spezifische oder generische Personen, Sachen oder Orte bezogen.

[Fid97] trennt basierend auf seinen Ergebnissen zur Analyse von 100 Anfragen die Suche nach Bildern nach den Daten- und Objektpolen. Zum Ersten zählen Systeme zum Retrieval von medizinischen Aufnahmen, chemischen Strukturen oder kartographischen Materialien. Bei diesen Bildern ist es irrelevant, wer die Bilder erstellt hat bzw. in welchem Bezug die Inhalte zu anderen Bildern oder Texten stehen. Sie dienen allein als Information. Das andere Ende bildet der sog. Objektpol, wobei Bilder ausschließlich als visuelle Ergänzung zu einem Produkt (Buch, Broschüre, Werbung) verwendet werden. Bilder werden dabei nur wegen einem Objekt, Ereignis oder einer abgebildeten Person gesucht und eingesetzt. Zwischen diesen zwei Polen befinden sich Künstler, Kunsthistoriker und Lehrkräfte, die Bilder sowohl wegen der abgebildeten Information als auch wegen der beinhalteten Objekte benötigen.

In [Jör98] wurden drei Untersuchungen mit 82 Teilnehmern in verschiedenen Gruppen durchgeführt. Die eine Gruppe musste die gezeigten Bilder allein nach dem Kriterium, was sie im Bild sehen, textuell beschreiben. Die Aufgabe der zweiten Gruppe war es die gezeigten Bilder möglichst so zu beschreiben, wie sie bei einer Suche in einer Bilddatenbank das gegebene Bild finden könnten. Die dritte Untersuchung hat ermittelt, an welche Beschreibungen sich die Testpersonen nach 4 Wochen noch erinnern konnten. Die von den Testpersonen verwendeten Attribute wurden 3 Haupt- und 12 Unterkategorien zugeordnet. Die 3 Hauptkategorien sind die wahrnehmenden (*p*), interpretativen (*i*) und reaktiven (*r*) Attribute, die in Abschnitt 3.2.2 näher beschrieben sind. Die 12 Unterkategorien sind literale Objekte (*p*), Personen (*p*), „Qualitäten“ von Personen¹ (*i*), kunsthistorische Information (*i*), Farbe (*p*), Ort (*p*), visuelle Elemente (*p*), Größen- bzw. Quantitätsbeschreibung (*p*), abstrakte Konzepte (*i*), Inhalt bzw. Geschichte (*i*), persönliche Reaktion (*r*) und externe Beziehungen. Auf eine genaue Beschreibung der Unterkategorien wird hier nicht weiter eingegangen, Details sind in [Jör98] zu finden. Im Schnitt ergab sich aus allen drei Untersuchungen, dass die Beschreibungen zu 32,8% aus Objekten, 8,6% aus inhaltlichen oder geschichtlichen Informationen, 8,5% aus Personen, 6,8% aus Farbinformationen und zu 5,7% aus Ortsangaben bestanden. Die restlichen

¹ z. B. Beziehungen zwischen Personen, mentale oder emotionale Zustände, Berufe

3. Annotation von Bildern

Attribute sind wegen ihrer niedrigen prozentualen Anteile vernachlässigbar.

Eine ähnliche Untersuchung bzgl. der Beschreibung von Bildern wurde in [JSW11] mit 35 Teilnehmern wiederholt. Von den Benutzern wurden insgesamt 7 192 Fotoaufnahmen aus dem Flickr Fotostream des Library of Congress (vorwiegend journalistische Aufnahmen) mit Termen versehen. Diesmal bestanden die Bildbeschreibungen im Durchschnitt zu 38,52% aus Objekten, 30,17% aus inhaltlichen oder geschichtlichen Informationen, sowie aus je 7,18% kunsthistorischen Informationen und „Qualitäten“ von Personen. Der hohe Anteil an inhaltlichen und geschichtlichen Termen ist durch die Auswahl der Bildquelle begründbar.

Die Art, wie Personen über Bilder und deren Inhalt denken, ist für die Annotation der Bilder sehr wichtig, da diese mit den Suchterminen vereinbar sein müssen. Jedoch kann die Interpretation von Bildern sehr unterschiedlich sein und unter den Betrachtern stark variieren. In [GO02] wurde die Perspektive der Benutzer untersucht, wie sie Bilder und deren Inhalte beschreiben. Die Forscher haben im Vorfeld 7 Kategorien zur Beschreibung von Bildern basierend auf Erkenntnissen der Literatur identifiziert: Farbe, Form, Textur, Personen bzw. Dinge, Ort, Handlung und Beeinflussung. Die Benutzer mussten 26 Wörter aus diesen 7 Kategorien 10 Bildern über Bäume und Landschaften zuordnen. Leider kann durch die niedrige Anzahl der Bilder die Aussagekraft der Untersuchung in Frage gestellt werden. Erkenntnisse waren jedoch, dass für die verwendeten Bilder von Bäumen und Landschaften die Farben oft, Formen jedoch nur sehr selten als Beschreibungsmerkmal identifiziert wurden. Außerdem wurde in dieser Studie noch herausgestellt, dass die Betrachter häufig Dinge sehen, die im Bild nicht direkt oder überhaupt nicht vorhanden sind. Diese Beschreibungen automatisch aus dem Bildinhalt zu extrahieren liegt nah an der Grenze des Unmöglichen.

In [Che01] wurde untersucht, inwieweit die Suchbegriffe im kunsthistorischen Umfeld den vorgeschlagenen Kategorien von [Ens93], [Jör98] und [Fid97] zuordenbar sind. Hierzu wurden 29 Studenten befragt, welche Suchterme sie sich zu Themen überlegt und welche Terme sie später tatsächlich zur Suche nach Bildern verwendet haben. Die angegebenen Begriffe wurden anschließend Kategorien zugeordnet bzw. diese Zuordnung ausgewertet und mit den Ergebnissen der ursprünglichen Arbeiten verglichen. Basierend darauf wurde ein Vorschlag zur Verbesserung der Kategorien in [Ens93] und [Jör98] vorgestellt. Eine weitere Erkenntnis der Studie war, dass in den Anfragen hauptsächlich spezifische Terme benutzt wurden; Farb-, Form- oder Texturinformationen waren jedoch rar. Am häufigsten kamen Personen, Orte und Objekte als Terme in den Anfragen vor.

In [CR03] ist eine sehr gute Übersicht zu Arbeiten bzgl. der Bedeutung und der Suche nach Bildern zu finden. Im zweiten Teil wurden Anfragen zu einer Bilddatenbank der US-Geschichte untersucht. Es handelt sich hierbei lediglich um 38 Anfragen, die 4 Kategorien zugeordnet wurden:

- spezielle Bedürfnisse, deren Terme sich auf eine Person, ein Ereignis oder eine Aktivität beziehen (z. B. „Thomas Jefferson“);
- generelle oder benennbare Bedürfnisse, die als Schlüsselwörter ausdrückbar sind (z. B. „eine Burgruine“);
- generelle oder abstrakte Bedürfnisse, die eher abstrakte Objekte enthalten (z. B. „eine Straße mit viel Verkehr“);
- subjektive Bedürfnisse, die emotionale Wirkungen ausdrücken (z. B. „eine Szene die den Wandel der Zeit illustriert“).

Die meisten Suchanfragen (mehr als 85%) fielen in die ersten beiden Kategorien. Im Durchschnitt wurden 4,87 Suchterme pro Anfrage angegeben, was deutlich mehr ist als der in [JSS00] und [JS05] zu normalen Websuchanfragen ermittelte Wert (2,4). Anfragen zu abstrakten Bedürfnissen beinhalteten im Schnitt sogar 7,7 Terme, subjektive 5,5.

In [HSWW04] wurden 30 Personen gebeten zu 3 ausgewählten Texten ein passendes Bild zu suchen. Zuerst mussten die Probanden das vorgestellte Bild textuell beschreiben und anschließend mittels der Bildsuche von Altavista über 5 Anfragen ein passendes Bild finden. Aus diesen Aufgaben ergaben sich insgesamt 180 Beschreibungen, die basierend auf der in Abschnitt 3.2.2 vorgestellten Struktur ausgewertet wurden. 87% der Anfrageterme gehörten demnach zur konzeptionellen Ebene, 12% zur Wahrnehmungs- und nur knapp 1% zur nichtvisuellen Ebene. Letzteres wurde auch durch die Auswahl der Texte beeinflusst. Auf der Wahrnehmungsebene waren die häufigsten Terme objektbezogen die Farbe und Zusammenstellung, szenenbezogen die Farbe und Typ bzw. Technik. Auf der konzeptionellen Ebene wurden zu 70% Objekte genannt, während sich 30% der Terme auf die komplette Szene bezogen. Generische Terme machten auf der konzeptionellen Ebene knapp 75% aus. Im Vergleich zwischen der freien Beschreibung der vorgestellten Bilder und den tatsächlichen Anfragen wurde beobachtet, dass bei den Suchanfragen spezifischere Terme verwendet wurden, was die Genauigkeit des Suchergebnisses erhöhte.

In [SZR00] wurde das Ausmaß der positiven Wirkung der Suche in verschiedenen Modalitäten untersucht. Das Informationsbedürfnis eines Benutzers wurde auf mehrere Anfragekomponenten unterteilt: Text, einfache Bildmerkmale (z. B. Farbhistogramme),

3. Annotation von Bildern

Objektkategorien, räumliche Beziehungen, Bildattribute (z. B. innen / außen) und Metadaten (z. B. Zeitstempel). Nachfolgend wurde versucht diese einzelnen Komponenten sowohl aus den Beschreibungstexten als auch aus den Anfragen zu ermitteln. In einem ersten Schritt wurden mittels regulärer Ausdrücke Objekte im Text identifiziert. Anschließend wurden basierend auf den ermittelten Objekten kurze Ausdrücke extrahiert und 5 Kategorien zugeordnet. Bei der Suche auf textueller Ebene wurde die semantische Distanz von WordNet [Mil95] zwischen den Objekten in den Anfragen und den Objekten in der Beschreibung verwendet. Auf einer zweiten Ebene wurden die Beziehungsvektoren untersucht. Auf Bildebene wurden Farbhistogramme in Verbindung mit der Gesichtserkennung benutzt, da hauptsächlich Personen und Landschaften erkannt werden sollten. Bei der Evaluation wurden 46 Anfragen an eine Datenbank mit 2 300 Bildern aus diversen Bereichen gestellt. Die Auswertung hat gezeigt, dass die Kombination von Text und Bildinhalten die Precision erhöht, jedoch den Recall senkt (vgl. Abschnitt 4.1.4.1). Nach den Erkenntnissen der Untersuchung sollte eine Zwischenstufe der Beschreibung (d. h. sowohl textuell als auch visuell) angestrebt werden.

In [CZJ07] und [ZCJ05] wurde der Nutzen des User Term Feedbacks beim interaktiven textbasierten Bildretrieval untersucht. Bisherige Untersuchungen suggerierten, dass durch die Möglichkeit Anfrageterme iterativ hinzuzufügen bzw. wegzunehmen, Benutzer in der Lage sind mehr und genauere Terme zu identifizieren, die relevant für das gesuchte Bild sind. Dabei können auch irrelevante Terme ausgewählt werden, was die iterative Suche verzögern oder sogar in die Irre führen kann. Das Szenario in dieser Untersuchung war ausgehend von den manuell erstellten Textannotationen der Bilder iterativ ein vorgegebenes Bild durch textbasierte Anfragen zu finden. Dabei wurden nach jedem Suchvorgang die jeweils 20 besten Ergebnisse angezeigt. Anschließend konnten die Anfragen entweder manuell oder durch automatisch vorgeschlagene Terme angepasst werden. Die vorgeschlagenen Terme wurden hierbei durch verschiedene Verfahren aus den Annotationen der gefundenen Bilder entnommen. Der Vergleich beider Anpassungsverfahren zeigt, dass User Term Feedback zwar die Brücke zwischen dem Vokabular des Benutzers und der Annotationen in der Datenbank schlägt, jedoch auch die Auswahl von irrelevanten Termen vorantreibt. Letzteres überwiegt leider, weswegen der Einsatz von User Term Feedback zur Zeit als sinnlos erscheint.

In [GS01] wurden 33 149 Benutzeranfragen an den Webbildsuchdienst Excite¹ untersucht. Im Schnitt wurden von den Benutzern 3,36 Anfragen pro Session gestellt, welche durchschnittlich 3,74 Terme beinhaltet haben. Die Terme in den Anfragen waren sehr unterschiedlich, die meisten Terme traten nur ein einziges Mal auf. Eine Kategorisierung der Anfrageterme nach [Jör98] oder [Ens93] wurde nicht durchgeführt, weswegen keine Aussage über die Verteilung der Terme bzgl. Objekte oder Personen möglich ist.

[SJWS02] bietet einen kurzen Überblick und einige Messergebnisse zur Veränderung der Websuche zwischen 1997 und 2001 basierend auf verschiedenen Untersuchungen auf der Excite Websuchmaschine. In [OSO03] wurden die Veränderungen der Suche nach multimedialen Inhalten mittels der Excite Suchmaschine zwischen 1997 und 2001 untersucht. Die Haupterkenntnis der Untersuchungen ist, dass sich zwar die Länge der Suchsessions von 1997 auf 2001 verkürzt, die Anfragen sich jedoch verlängert haben. Auch boolesche Operatoren werden immer häufiger eingesetzt.

In [JSP03] wurden die Ergebnisse des textbasierten Suchverhaltens nach multimedialen Inhalten von Benutzern von AltaVista² untersucht. Berücksichtigt wurden 369 350 Sessions mit insgesamt 1 073 388 Anfragen. Die zwei hauptsächlichen Fragen der Untersuchung waren:

- Was sind die Charakteristika der Suche nach multimedialen Inhalten bei AltaVista?
- Wie sieht die Suche nach multimedialen Inhalten im Vergleich zur normalen Web-Suche aus?

Es wurde ermittelt, dass bei der Suche nach multimedialen Inhalten eine wesentlich höhere Interaktion zwischen Benutzern und Suchmaschine benötigt wird. Dies begründet sich durch die Erhöhung der Anfragen- und Sessionlängen sowie der Anzahl der Ergebnisseiten, welche angeschaut wurden. Eine klare Richtung für die Suche konnte nicht gefunden werden. Die hohe Diversität der Suchanfragen zeigt, dass die Benutzer nach einer stetig steigenden Anzahl von Themen suchen. Die Suchterme wurden in dieser Arbeit nicht klassifiziert. Die Anzahl der Anfrageterme für die Bildsuche war im Schnitt deutlich höher als bei der Suche nach Audio oder Video (4 Terme im Gegensatz zu weniger als 3 Termen pro Anfrage). Die Länge der Sessions war bei den Bildern am größten, dicht gefolgt von der Suche nach Videos. Bildsuchen zeichnen sich auch dadurch aus, dass die Benutzung von booleschen Operatoren hier am höchsten war.

1 <http://www.excite.com/>

2 <http://www.altavista.com/>

3. Annotation von Bildern

In [Pu05] wurden 2,4 Millionen Anfragen an den chinesischen Webbildsuchdienst VisionNEXT¹ ausgewertet. Insgesamt wurden in diesen Anfragen mehr als 325 812 unterschiedliche Suchbegriffe verwendet. Die Anfragen wurden auch daraufhin untersucht, ob etwas Spezifisches (z. B. „Tower Bridge“) oder etwas Generelles (z. B. „Brücke“) gesucht wurde, bzw. ob die Suche durch die zusätzliche Angabe von einem Ort oder einem Zeitpunkt verfeinert wurde. 75,85% der Anfragen an den Suchdienst waren auf spezielle, 17,91% auf generische Personen, Orte oder Objekte ausgerichtet. Die restlichen Anfragen waren Verfeinerungen dieser. Daraus lässt sich ableiten, dass bei der Bildsuche im Internet die Anfragen zu 93,76% auf Personen, Orte oder Objekte ausgerichtet sind. Als Fortsetzung der Studie aus [Pu05] wurden in [Pu08] erfolgreiche und fehlgeschlagene Suchanfragen von VisionNEXT evaluiert. Fehlgeschlagene Suchanfragen waren im Schnitt länger (4,12 chinesische Schriftzeichen) als erfolgreiche (2,83 chinesische Schriftzeichen). Wesentlich ist auch der Unterschied bzgl. der Häufigkeit der Anfragen. 95% der erfolgreichen Anfragen wurden mehr als 3 Mal gestellt, während 57,78% der fehlgeschlagenen Anfragen weniger als 3 Mal vorkamen. Letzterer hoher Wert ist ein Indiz für die Einzigartigkeit und Spezifität der Anfragen, welche letztendlich fehlschlagen.

[JJ05] betrachteten das Suchverhalten von professionellen Benutzern. Hierzu wurde die Logdatei eines kommerziellen Bildanbieters im Zeitraum von einem Monat evaluiert. Die durchschnittliche Anzahl der Anfragen je Session lag bei 2,1, wobei je Anfrage im Schnitt 1,87 Terme angegeben wurden. Mehr als die Hälfte der Anfrageterme waren Objekte, was u. a. auch durch den verwendeten Datenbestand erklärt werden kann.

[TSJ09] untersuchte die Veränderungen der Multimedia Websuche zwischen 1997 und 2006 anhand insgesamt 1 228 330 Anfragen in 361 319 Sessions der Metasuchmaschine Dogpile². 54,5% der betrachteten Sessions – und somit die überwiegende Mehrheit – diente der Suche nach Bildern. Im Durchschnitt wurden 4,9 Minuten je Session für die Bildsuche aufgewendet, 2,8 Anfragen je Session gestellt und 2,22 Terme je Anfrage angegeben. Die Suchanfragen dienten zu 56,2% zur Auffindung von Bildern zu Personen, zu 21,7% für Objekte und Szenen und zu 7,4% für medizinische Sachverhalte.

In [HTV11] wurden insgesamt 1 094 620 Bildsuchanfragen an das Portal einer europäischen Nachrichtenagentur untersucht. Je Session wurden im Durchschnitt 2,1 Anfragen gestellt, die Anzahl der Terme je Anfrage betrug im Schnitt 2,25. Am häufigsten wurde

1 <http://www.visionnext.com/>

2 <http://www.dogpile.com>

mit 44% der Anfragen nach spezifischen Personen gesucht, was durch den verwendeten Datensatz erklärt werden kann. An zweiter Stelle folgten mit 12,3% der Anfragen spezifische Orte und mit 6,6% Objekte.

In [Cho10] wurde das Suchverhalten von 29 Studenten bei der Suche nach Bildern im Web untersucht. Die Teilnehmer mussten Bilder zu ihren Seminarprojekten, Bilder zu ihrer (projektunabhängigen) Arbeit sowie Bilder zu ihren persönlichen Interessen suchen. Bei der Evaluation wurde festgestellt, dass zur Lösung einer Suchaufgabe bei allgemeinen Suchmaschinen im Durchschnitt 4,2 Anfragen je Session notwendig waren, während bei Bildsuchmaschinen dieser Wert 5,55 und auf lokalen Bilderseiten 1,49 betrug. Auch die Länge der Anfragen war unterschiedlich. Die durchschnittliche Länge von Anfragen bei allgemeinen Suchmaschinen betrug 3,46 Terme, bei Bildsuchmaschinen 3,37 Terme und auf lokalen Bilderseiten 2,23. Die Bildsuchen für Seminarprojekte bestanden im Durchschnitt aus 3,29 Termen, während für die Bildsuche für sonstige Arbeitsbereiche 2,98 Terme und für persönliche Interessen 3,29 Terme verwendet wurden.

Basierend auf den Erkenntnissen von den Text Retrieval Wettbewerben Text REtrieval Conference (TREC), AMARYLLIS und Cross Language Evaluation Forum (CLEF), sowie ausgehend von Interviews mit Nutzern von Bildretrieval Systemen haben [FMH06] fünf wesentliche Aufgaben innerhalb des Bildretrievals identifiziert:

- Auffinden von urheberrechtlich geschützten Bildern, welche ggf. durch Bildtransformationen leicht verändert wurden.
- Suche nach Bildern zur Illustration von Texten ausgehend von einer textuellen Anfrage.
- Erkennung und Transkription von Text in Bildern.
- Identifikation von Objekten bzw. Objektklassen zur Indizierung und Filterung von Bildern.
- Beschreibung des allgemeinen Kontexts von Bildern, wie z. B. Innen- oder Außen-, Tag- oder Nacht-, Stadt- oder Landschaftsaufnahmen.

In [CAG00, EBB04] wurde das textbasierte Suchverhalten von 45 Benutzern professioneller Bildretrievalssysteme untersucht. Leider war das Suchverhalten der einzelnen Teilnehmer der Studie viel zu unterschiedlich. Bezogen auf die Verwendung von Bildern konnten jedoch 7 Klassen identifiziert werden:

- Illustration in Verbindung mit anderen Medien (wie z. B. bei Nachrichten),
- Informationsverarbeitung (Relevanz der Daten im Bild, z. B. Röntgenaufnahmen),
- Verteilung bzw. Übermittlung von Information,
- Lernen,
- Erzeugung von Ideen bzw. als Inspirationsquelle,
- gefühlsbetonter, überzeugender Einsatz (z. B. in der Werbung),
- ästhetische Zwecke (Verwendung der Bilder als reine Dekorationselemente).

Zusammenfassung

In den Tabellen 3.1 und 3.2 sind die wesentlichen Attribute, mit denen Bilder textbasiert gesucht bzw. durch welche Bilder beschrieben wurden, sowie weitere Erkenntnisse aus den vorgestellten Arbeiten zusammengefasst. Unabhängig davon, ob das Web oder kunsthistorische bzw. journalistische Bilddatenbanken zum Einsatz kommen, kann festgestellt werden, dass hauptsächlich nach Personen, Objekten und Orten gesucht wird. Farben, Formen und Texturen sind selten unter den Suchbegriffen aufzufinden. Die durchschnittliche Länge der Suchanfragen variiert zwischen 3 und 5 Termen, wobei es sich sowohl um spezifische als auch um generische Begriffe handelt.

3.2.2 Logische Strukturierung der Attribute von Bildern

In den vergangenen Jahrzehnten wurde eine Vielzahl an Merkmalen und Metadaten konzipiert, die das Ziel verfolgen, den Inhalt eines Bildes genau und möglichst kompakt zu repräsentieren. Auch um die logische (bzw. mehr oder weniger philosophische) Klassifikation von diesen Bildattributen haben sich bereits mehrere Studien bemüht. In diesem Abschnitt werden die Ansätze zur logischen Strukturierung der Attribute eines Bildes vorgestellt.

Eine der ersten Arbeiten zur Struktur für Beschreibungen von Bildern stammt von [Pan57]. In diesem Buch wurden drei verschiedene Ebenen identifiziert: die prä-ikonographische, die konventionelle und die subjektive Interpretationsebene. Auf der ersten, prä-ikonographischen Ebene werden identifizierbare Objekte, Personen oder Ereignisse als solche mit ihrer natürlichen Bedeutung wahrgenommen. Auf der konventionellen Ebene werden basierend auf der Handlung und den Gesten im Bild weitere Bedeutungen

QUELLE	KATEGORIE	Typ	ATTRIBUTUM	SONSTIGE ERKENNTNISSE
[AE97]	Zeitung	Suche	spez./gen. Personen, Sachen, Orte	
[CAG00]	diverse	Suche	–	Identifikation von 7 Klassen für die Verwendung von Bildern
[Che01]	Kunst	Beschreibung, Suche	Personen, Objekte, Orte	hauptsächlich spezifische Terme in den Anfragen
[Cho10]	Web	Suche	–	durchschnittlich 4,2 bzw. 5,55 Anfragen je Session bei allgemeinen bzw. Bildsuchmaschinen, 2,23 bis 3,46 Terme je Anfrage
[CR03]	Geschichte	Suche	Personen, Objekte, Ereignisse	4,87 Terme pro Anfrage für Bildsuche
[CZJ07]	Web	Suche, Feedback	–	User Term Feedback hat mehr negative als positive Auswirkungen
[Ens93]	Zeitung	Suche	Personen, Objekte, Ereignisse	
[FMH06]	diverse	Suche	Kopien, Text, Objekte	Identifikation von 5 Suchanfragetypen
[Fid97]	Zeitung	Suche	abhängig vom Pol	Kategorisierung der Benutzergruppen, Daten- und Objektpol
[GO02]	Natur	Beschreibung	Farben oft, Formen selten	Betrachter sehen oft Dinge die nicht direkt oder überhaupt nicht im Bild enthalten sind
[GS01]	Web	Suche	–	3,36 Anfragen pro Session, 3,74 Terme pro Anfrage
[HSWW04]	Web	Beschreibung, Suche	Terme auf konzeptioneller Ebene und generische Terme	bei Suche Tendenz zu spezifischeren Termen im Gensatz zur Beschreibung

Tabelle 3.1: Kategorisierung der Veröffentlichungen zur textbasierten Suche nach Bildern – Teil 1.

3. Annotation von Bildern

QUELLE	KATEGORIE	Typ	ATTRIBUTE	SONSTIGE ERKENNTNISSE
[HTV11]	Zeitung	Suche	spezifische Terme, vor allem Personen, Orte und Objekte	2,1 Anfragen pro Session, 2,25 Terme pro Anfrage
[JJ05]	Web	Suche	vorwiegend Objekte	2,1 Anfragen pro Session, 1,87 Terme je Anfrage bei professionellen Nutzern
[Jäi98]	Kunst	Beschreibung, Suche	Objekte, inhaltliche oder geschichtliche Informationen, Personen, Farben, Orte	
[JSP03]	Web	Suche	–	bei Bildsuche verwenden Benutzer längere Sessions, mehr Suchterme und mehr boolesche Operatoren als bei Audio- oder Videosuche, 4 Terme pro Anfrage
[JSW11]	Geschichte	Beschreibung	Objekte, inhaltliche oder geschichtliche Informationen	
[Orn95]	Zeitung	Suche	bekannte Personen	Einteilung der Journalisten in 5 Benutzergruppen
[OSO03]	Web	Suche	–	kürzere Sessions, längere Anfragen, mehr boolesche Operatoren im Vergleich vom Jahr 1997 auf 2001
[Pu05]	Web	Suche	Personen, Objekte, Orte, spezifische Terme	
[Pu08]	Web	Suche	–	fehlgeschlagene Suchanfragen sind im Durchschnitt länger und spezifischer bzw. einzigartiger als erfolgreiche Anfragen
[SZR00]	diverse	Suche	Personen, Orte	Suchergebnis präziser, wenn text- und bildbasierte Suche bzw. Beschreibungen verbunden werden
[TSJ09]	Web	Suche	Personen, Objekte und Szenen, medizinische Sachverhalte	4,9 Minuten je Session, 2,8 Anfragen pro Session, 2,22 Terme pro Anfrage

Tabelle 3.2: Kategorisierung der Veröffentlichungen zur textbasierten Suche nach Bildern – Teil 2.

wahrgenommen. Die dritte Ebene reichert das Bild basierend auf der Belesenheit, dem Wissen und dem kulturellen Hintergrund des Betrachters mit zusätzlicher, teilweise subjektiver Bedeutung an. Diese drei Ebenen wurden in [Lin68] um eine empirische Ebene erweitert, bei dem zum Bild noch vermerkt werden kann, wann, wer, wo und warum das Bild bzw. das Kunstwerk geschaffen hat.

[Mar88] greift die drei Ebenen von [Pan57] auf und ordnet die zu dieser Zeit existierenden Systeme in eine eigens konstruierte Matrix ein. Die ersten zwei Ebenen werden in dieser Arbeit als primäre (Form, Farbe, Textur) und sekundäre Thematik (Thema, Geschichte, Konzepte) bezeichnet.

[Kra88] teilt die Bedeutung von Bildern ähnlich auf zwei Ebenen auf. Die sog. harte Indizierung beschreibt objektiv, was im Bild zu sehen ist, während sich die sog. weiche Indizierung auf die subjektive persönliche Bedeutung sowie auf die Nachricht hinter dem Bild bezieht.

[SL94] hat ihre Kategorisierung von textuellen Bildattributen bzgl. kunsthistorischer Bilder ebenfalls auf die Aufteilung von [Pan57] aufgebaut. In [SL94] werden biografische, themenspezifische, beispielhafte und Beziehungsattribute unterschieden.

Biografische Attribute entsprechen im Wesentlichen der empirischen Ebene von [Lin68]: sie beschreiben die Entstehung des Bildes, d. h. wer, wann, wo und warum das Bild bzw. das Kunstwerk geschaffen hat. Des Weiteren gehören zu den biografischen Attributen auch die Informationen, wo das Bild heute zu finden ist, wo es bislang untergebracht war, wem es gehört, wie viel es wert ist und ähnliche Metainformationen.

Themenspezifische Attribute können auf drei Unterebenen aufgeteilt werden. Die erste Ebene beschäftigt sich damit, was das Bild enthält („of“) und worüber das Bild handelt („about“). Zum Beispiel kann in einem Bild ein Mann und ein Löwe zu sehen sein, wobei das Bild über Hochmut handelt. Auf der zweiten themenspezifischen Ebene sind Bilder in den meisten Fällen gleichzeitig generisch und spezifisch. Zum Beispiel kann auf einem Bild die Tower Bridge zu sehen sein (spezifisch), aber es kann auch nur als eine Abbildung einer Brücke (generisch) aufgefasst werden. Die dritte themenspezifische Ebene unterscheidet die Bildinhalte nach Raum, Zeit, Aktivitäten bzw. Ereignissen und Objekten. Bilder können demnach spezifisch einen oder vier Aspekte enthalten bzw. generisch über diese Aspekte handeln oder diese enthalten.

Beispielhafte Attribute versuchen eine gewisse Metaebene im Bild festzuhalten. Zum Beispiel kann ein Bild ein Poster oder eine Postkarte sein, es kann aber auch ein Bild mit Postern und Postkarten sein, wobei es dann als Beispiel dient.

3. Annotation von Bildern

Zuletzt werden in [SL94] noch Beziehungsattribute identifiziert, die Assoziationen zu anderen Bildern, Texten oder Objekten beschreiben. Als Beispiel können Zeichnungen zu literarischen Werken dienen. In [Fid97] wird die gleiche Einteilung der Bildattribute verwendet.

[Jör96] ordnet Bildattribute drei Gruppen zu. Wahrnehmende („perceptual“) Attribute sind direkte Beschreibungen des Bildes, wie z. B. die Farbe „rot“ oder das Objekt „Auto“, welche im Bild auftauchen. Interpretierende („interpretive“) Attribute erfordern eine Interpretation der wahrnehmbaren Hinweise durch ein gewisses Hintergrundwissen. Solche Attribute beschreiben z. B. den Stil („moderne Kunst“) oder die Atmosphäre des Bildes („verträumt“). Reaktive („reactive“) Attribute halten persönliche Reaktionen bzgl. des Bildes fest, wie z. B. „Verwirrtheit“ oder ob dem Betrachter das Bild gefällt.

Zwar wurden in [AE97] Benutzeranfragen untersucht und in zwölf nicht-disjunkte Mengen eingeordnet, jedoch kann die resultierende Aufteilung auch zur Kategorisierung von Bildattributen verwendet werden. Die Anfragen wurden basierend auf der Verbindung der Erkenntnisse von [Pan57] und [SL94] in eine Matrix eingeordnet. Die zwölf Mengen sind in der Tabelle 3.3 dargestellt.

	PRÄ- IKONOGRAPHISCHE EBENE (GENERISCH)	KONVENTIONELLE EBENE (SPEZIFISCH)	SUBJEKTIVE INTER- PRETATIONSEBENE (ABSTRAKT)
WER?	Art der Person oder des Objekts	individuell benannte Person, Gruppe, Objekt	mythologisches oder fiktives Wesen
WAS?	Art des Ereignisses, Zustands	individuell benanntes Ereignis	Emotionen oder Abstrahierungen
Wo?	Art des Ortes: geografisch, architektonisch	individuell benannter geografischer Ort	symbolisierter Ort
WANN?	zyklische Zeit: Jahreszeit, Tageszeit	lineare Zeit: Datum oder Zeitdauer	durch Zeit symbolisierte Emotionen oder Abstraktionen

Tabelle 3.3: Panofsky-Shatford Matrix nach [AE97].

[Eak98] stellt drei Ebenen zur Beschreibung von Bildern vor. Die erste Ebene umfasst dabei primitive Merkmale wie z. B. Farben, Formen und Texturen. Auf der zweiten Ebene werden basierend auf den Primitiven der ersten Ebene logische oder abgeleitete Elemente identifiziert (z. B. „Frau“, „Stuhl“, „Landschaft“). Auf dieser Ebene können

sowohl generische als auch spezifische Angaben gemacht werden (z. B. „eine Frau“ oder „Mona Lisa“). Die dritte Ebene erfasst Assoziationen und Einflüsse auf den Betrachter. In [GO02] wurden die drei Hauptkategorien ähnlich zu denen in [Eak98] definiert. Außerdem wurden sieben Unterkategorien von Attributen zur Beschreibung von Bildinhalten basierend auf den Erkenntnissen in der Literatur identifiziert. Getrennt werden Attribute nach primitiven Merkmalen (Farbe, Form, Textur), nach Objekten (Personen bzw. Dinge, Ort, Handlung) und nach der Beeinflussung des Betrachters (subjektives Empfinden).

[Bim99] identifiziert drei verschiedene Kategorien von Metadaten, welche mit Bildern assoziiert werden können. Zum einen können inhaltsunabhängige („content-independent“) Metadaten Bildern zugeordnet werden, wie z. B. das Datum und der Ort der Aufnahme oder der Name des Fotografen. Inhaltsbezogene Metadaten werden in zwei Gruppen aufgeteilt. Als inhaltsabhängige („content-dependent“) Metadaten werden Merkmale der niedrigen Ebene wie z. B. Farben, Formen oder Texturen bezeichnet. Inhaltsbeschreibende („content-descriptive“) Metadaten beziehen sich auf die Semantik des Bildes und fassen somit Objekte, Szenen, Ereignisse und Gefühle zusammen.

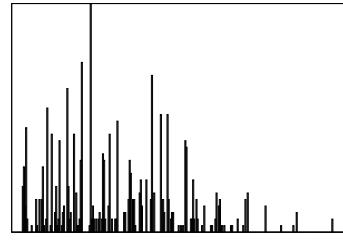
In [JC02] werden visuelle von nichtvisuellen Inhalten getrennt. Visuelle Inhalte sind demnach direkt wahrnehmbare Inhalte, wie z. B. Linien, Formen, Farben, Objekte; während nichtvisuelle Inhalte im Prinzip die Bildmetadaten umfassen, wie z. B. Name des Fotografen oder Besitzer eines Gemäldes. Die visuellen Inhalte werden auf 2 Haupt- und insgesamt 10 Unterebenen aufgeteilt. Die Syntax-Hauptebene bezieht sich auf die direkte Wahrnehmung der Bildinhalte im Sinne von Reflexionen des Lichts. Dies umfasst z. B. die wahrgenommenen Farben und Texturen ohne jegliche Interpretation. Die Syntaxebene wird weiter unterteilt auf den Typ bzw. die Technik des Bildes, die globale Verteilung, die lokale Struktur und die globale Komposition. Auf der Semantik-Hauptebene werden generische, spezifische und abstrakte Konzepte (Objekte und Szenen) wahrgenommen. Beispiele für die einzelnen Ebenen sind in Abbildung 3.1 dargestellt.

[BBE03] klassifiziert Bildinhalte in 9 Kategorien. Wahrnehmende Primitive beziehen sich auf die niedrigste Ebene des Bildinhalts, wozu Farben und einfache Texturbeschreibungen gehören. Geometrische Primitive sind einfache zwei- oder dreidimensionale Formen, die selber noch keine Bedeutung tragen, wie z. B. Linien, Kreise, Rechtecke. Die sog. visuelle Erweiterung erfordert schon ein gewisses Maß an Inferenz. Hierzu gehören z. B. die Ermittlung der Tiefe basierend auf Schatten oder der Perspektive. Semantische Einheiten entsprechen generischen und spezifischen Namen, die den Inhalt benennen. Die kontextuelle Abstraktion bildet universelle Assoziationen oder Interpretationen basierend

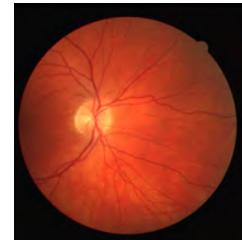
3. Annotation von Bildern



(a) Typ, Technik (HDR Farbfoto)



(b) globale Verteilung (Farbhistogramm)



(c) lokale Struktur (Augenuntersuchung)



(d) globale Komposition (lineare Anordnung)



(e) generische Objekte (Enten)



(f) generische Szenen (Außenbereich, Landschaft)



(g) spezifische Objekte (Il Commendatore von Anna Chromy)



(h) spezifische Szenen (Budapest)



(i) abstrakte Objekte (Kunst, Theater)



(j) abstrakte Szenen (Montag Morgen)

Bild 3.1: Beispielbilder für die Attributebenen nach [JC02].

auf dem Allgemeinwissen. Durch kulturelle Abstraktion werden Assoziationen identifiziert, die auf kulturellen Eigenheiten basieren. Technische Abstraktion beruht auf den Kenntnissen und Vokabularen von Experten. Die emotionale Abstraktion erfasst Einflüsse des Bildes auf den Betrachter, welche generalisierbar oder ausschließlich bezogen auf den Betrachter sein können. Zuletzt können Attribute eines Bildes noch der Kategorie der Metadaten zugeordnet werden, die unabhängig vom Bildinhalt sind, wie z. B. Bildgröße oder Format.

In [HSWW04] wurden die Attribute von Bildern auf drei Ebenen kategorisiert: die Wahrnehmungs-, die konzeptionelle und die nichtvisuelle Ebene.

Die Wahrnehmungsebene beinhaltet Informationen, die direkt aus dem Bild ableitbar sind. Im wesentlichen handelt es sich hierbei um Bildbestandteile, aus denen weitere Elemente zusammengesetzt werden können. Auch die Informationen zu Farben, Texturen und Formen gehören zu dieser Ebene.

Auf der konzeptionellen Ebene wird der Inhalt des Bildes beschrieben. Diese Ebene wird auf drei weitere Ebenen unterteilt: die generische, die spezifische und die abstrakte Sub-Ebene. Zur Beschreibung von generischen Konzepten ist lediglich Alltagswissen vonnöten, d. h. der Bildinhalt wird auf einer allgemeinen Ebene beschrieben (z. B. „ein Affe isst eine Banane“). Auf der Ebene der spezifischen Konzepte werden die Bildelemente genau benannt, d. h. im Beispiel wird der Affe z. B. durch seinen Namen, seinem Geburtsort und seinem aktuellen Aufenthaltsort beschrieben. Abstrakte Konzepte interpretieren den Bildinhalt und beschreiben ihn auf einer subjektiven Ebene.

Die nichtvisuelle Ebene befasst sich mit Daten bzw. Informationen, die nicht direkt aus dem Bild ermittelbar, jedoch objektiv sind. Dazu gehören z. B. Datum, Ort oder Zugriffsrechte.

Zusammenfassung

In Abbildung 3.2 sind die logischen Strukturierungen für Annotationen der vorgestellten Arbeiten zusammengefasst als Mindmap zu sehen. Tabelle 3.4 zeigt die Konzepte der Arbeiten nebeneinander gestellt, wodurch Synonymbeziehungen unter den verschiedenen Begriffen verdeutlicht werden. Die meisten vorgestellten logischen Strukturierungen trennen direkt aus den Bildern ermittelte Beschreibungen (Farben, Formen und Texturen), die daraus abgeleiteten visuellen Elemente (generische und spezifische Personen, Objekte und Orte sowie weitere Abstraktionen, wie z. B. Assoziationen, Gefühle) und nichtvisuelle Metadaten.

3. Annotation von Bildern

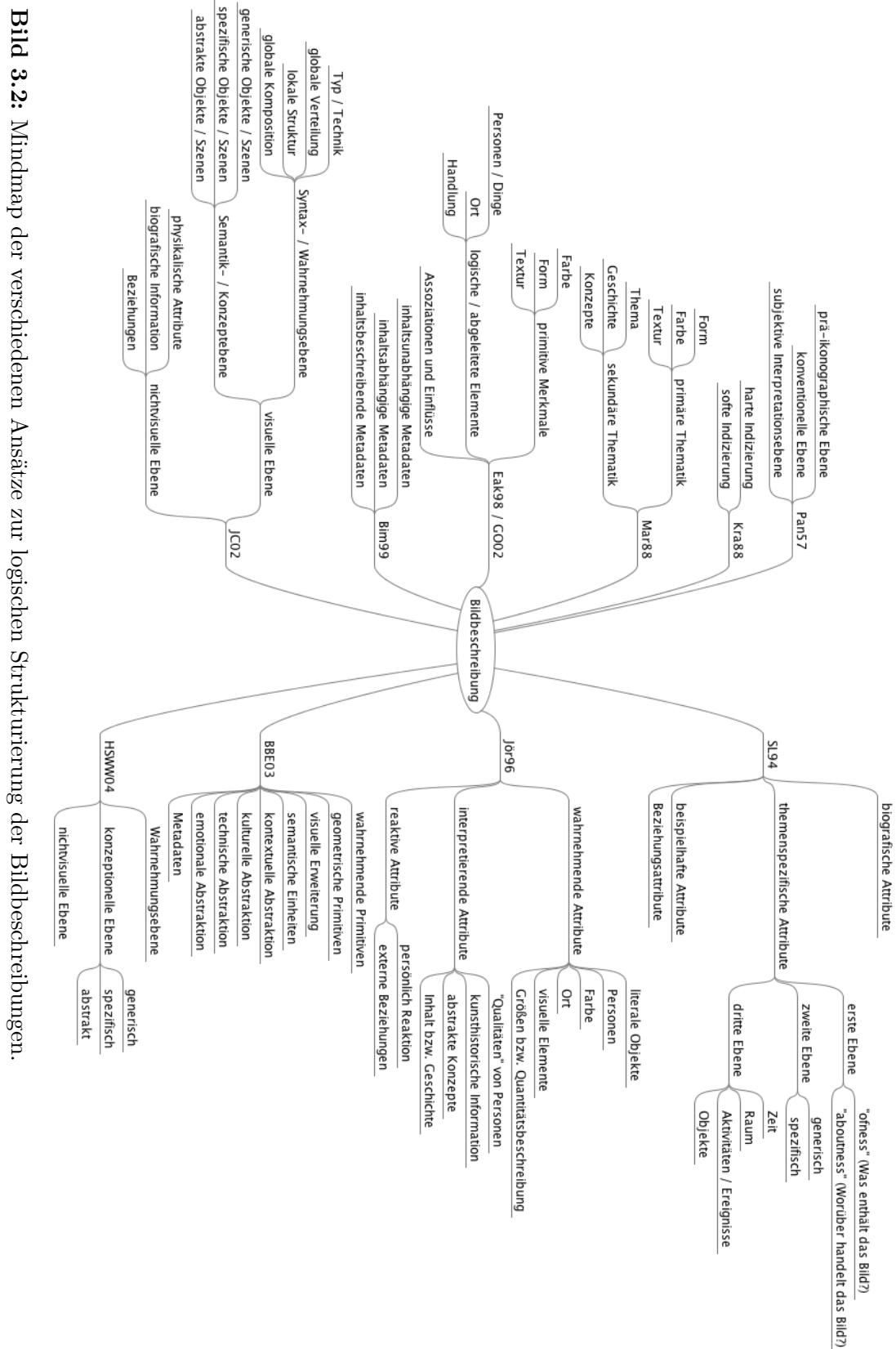


Bild 3.2: Mindmap der verschiedenen Ansätze zur logischen Strukturierung der Bildbeschreibungen.

[PAN57]	[MAR88]	[KRA88]	[SL94]	[JOR96]	[EAK98], [GO02]	[BIM99]	[JC02]	[BBE03]	[HSWW04]
primäre Thematik	Indizierung	harte Indizierung			prinzipielle Features	inhaltshängige Metadaten	Syntaxebene	wahrnehmende Primitiven geometrische Primitiven	Wahrnehmungsebene
präikonographische Ebene	sekundäre Thematik	softe Indizierung	themen spezifische Attribute	wahrnehmende Attribute	logische / abgeleitete Elemente	interpretierende Attribute	genetische Objekte / Szenen	visuelle Erweiterung semantische Einheiten	
konventionelle Ebene	subjektive Interpretations ebene				Assoziationen und Einflüsse	personliche Reaktion	spezifische Objekte / Szenen	kontextuelle Abstraktion	Konzeptionelle Ebene
empirische Ebene [Lin68]				beispielhafte Attribute	externe Beziehungen		nichtvisuelle Ebene	abstrakte Objekte / Szenen	kulturelle Abstraktion technische Abstraktion emotionale Abstraktion
				Beziehungsattribute	biografische Attribute	inhaltunabhängige Metadaten		Metadaten	nichtvisuelle Ebene

Tabelle 3.4: Vergleichende Tabelle der verschiedenen Ansätze zur logischen Strukturierung der Bildbeschreibungen.

3.2.3 Zusammenfassung

Für eine möglichst erfolgreiche textbasierte Suche müssen die textuellen Annotationen den Suchkriterien der Benutzer bzw. den Informationswünschen von sehbehinderten Personen entsprechen. Die in Abschnitt 3.2.1 und 3.2.2 betrachteten Arbeiten haben gezeigt, dass in kunsthistorischen Bildarchiven, journalistischen Bilddatenbanken und im Web in den meisten Fällen nach Objekten, Personen und Orten gesucht wird. Diese Anforderung deckt sich bis auf die Orte auch mit den Wünschen von sehbehinderten Menschen in Abschnitt 3.1.2. Zur Beschreibung und Suche von Bildern werden sowohl generische als auch spezifische Terme verwendet. Aus der Sicht der textbasierten Suche ergeben sich dadurch folgende Anforderungen für die automatische Annotation von Bildern:

- Primitive Merkmale, wie z. B. Farbe, Form und Textur finden bei der textbasierten Suche primär kaum Verwendung. Die daraus abgeleiteten logischen Elemente (z. B. Objekte, Personen) sind jedoch von hoher Relevanz.
- Objekte müssen erkannt und identifiziert werden. Die Benennung der Objekte ist sowohl generisch als auch spezifisch erwünscht. Eine grobe Zuordnung der Position des Objekts im Bild kann weitere Vorteile bei der Suche mit sich bringen, sofern die Suche Positionsangaben zulässt. Für die optimale Granularität der Objekterkennung muss eine Mindestgröße bzw. ein Maß für die Wichtigkeit des Objekts im Bild spezifiziert werden.
- Personen müssen ebenfalls erkannt und identifiziert werden. Die generische Benennung von Personen ist nötig, jedoch haben die spezifischen Namen der Personen ein höheres Gewicht.
- Ereignisse und Aktivitäten im Bild sollten nach Möglichkeit erfasst werden. Diese können Beziehungen zwischen Objekten und Personen beschreiben.
- Sofern möglich, sollten die zugehörigen Orte aus den Bildern ermittelt werden. Neue Technologien, wie z. B. mit GPS-Chips versehene Fotokameras und Handys sowie GPS-Zubehör (z. B. PhotoGPS¹) in Verbindung mit dem Dienst Geonames² schaffen hierbei Abhilfe und reichern die Bilder mit Metadaten bzgl. der Ortsinformation

1 <http://www.jobo.de/web/photoGPS.150.0.html>

2 <http://www.geonames.org/>

an. Interessant ist auch die Ergänzung von Kameras und Smartphones mit einem Kompass, wodurch auch die Aufnahmerichtung vermerkt werden kann.

- Abstrakte Konzepte sowie Eindrücke und Gefühle müssen nicht ermittelt und annotiert werden, da nur sehr wenige Anfragen darauf ausgerichtet sind.

3.3 Standards für Bildmetadaten und Bildannotationen

Im Laufe der Zeit sind für multimediale Inhalte verschiedene Metadatenstandards entstanden, welche neben strukturierten Metadaten auch Fließtext oder nicht-textuelle Informationen enthalten können. Neben der Standardisierung der Metadatenfelder wurden auch mehrere Konzept-Thesauri erstellt, um die Annotation von multimedialen Inhalten möglichst einheitlich zu gestalten. In Abschnitt 3.3.1 werden die gängigsten Schemata zur Abspeicherung von Bildmetadaten und in Abschnitt 3.3.2 die wichtigsten Konzept-Thesauri zur Annotation von Bildern kurz vorgestellt.

3.3.1 Standards für Bildmetadaten

In diesem Abschnitt werden die gängigsten Standards zur Speicherung von Bildmetadaten kurz vorgestellt. Die bekanntesten und wichtigsten Standards sind Dublin Core¹, MPEG-7 bzw. MPEG-21² [Kos04], Pro-MPEG³, DMS-1 bzw. EXF von SMPTE⁴, IPTC⁵, TV-Anytime⁶, VRA⁷, NISO [NIS06] und das P/Meta Schema von der European Broadcasting Union (EBU)⁸. [SS06] gibt einen kurzen, [BVB07] einen ausführlicheren Überblick über aktuelle Metadatenstandards. Eine genauere Analyse der verschiedenen Standards aus Sicht der Interoperabilität ist in [PVS08] zusammengefasst.

Die Media Annotations Working Group (MAWG)⁹ von der W3C verfolgt das Ziel alle gängigen Metadatenstandards für Multimedia (vorwiegend Videos) in einer Ontologie –

1 <http://dublincore.org>

2 <http://www.chiariglione.org/mpeg/>

3 <http://www.pro-mpeg.org>

4 <http://smpte.org>

5 <http://www.iptc.org>

6 <http://www.tv-anytime.org>

7 <http://www.vraweb.org/>

8 <http://www.ebu.ch>

9 <http://www.w3.org/2008/WebVideo/Annotations/>

die sog. Media Ontology bzw. Ontology for Media Resources – zusammenzufassen. Die Media Ontology definiert dabei sowohl ein Kernvokabular als auch die Abbildungen auf die Terme aller gängigen Metadatenstandards. Die Möglichkeiten für die Realisierung der Abbildungen sind in [SBB⁺09] beschrieben. Aktuell ist die Ontology for Media Resources als Technical Recommendation der W3C in der Version 1.0 erhältlich und stellt die Abbildungen zu den gängigen Metadatenstandards als RDF/OWL zur Verfügung.

Speziell für die Verarbeitung von Bildern, die mittels der digitalen Fotografie entstanden sind, haben sich in den letzten Jahren drei Metadatenstandards etabliert: Exif [EXI02] (bzw. TIFF 6.0 [TIF92]), IPTC-IIM [IPT99] und XMP [XMP08a, XMP08b, XMP08c]. Nachfolgend werden die einzelnen Standards kurz vorgestellt und die Möglichkeiten zur Abspeicherung von Inhaltsbeschreibungen erläutert.

3.3.1.1 Exif

Der Exif-Standard wurde von der Japan Electronics and Information Technology Industries Association (JEITA) entwickelt und liegt aktuell in der Version 2.2 vor [EXI02]. Der Standard ist in der digitalen Fotografie sehr weit verbreitet, da heute jede Digitalkamera beim Erstellen der Bilder Exif-Daten mit abspeichert und in die Bilder einbettet. Exif baut auf dem TIFF 6.0 [TIF92] Standard auf und verwendet auch einige der im TIFF Standard spezifizierten Felder. Dementsprechend werden die Metadaten bei Exif auch im Attribut-Wert-Format abgespeichert. In diesen Attributen werden sowohl Registrierungs- als auch Beschreibungsdaten verwaltet, die jedoch hauptsächlich Metadaten der Bilderrstellung und des Kamerasensors enthalten (z. B. Auflösung, Orientierung, Belichtungszeit, Brennweite, ISO-Empfindlichkeit, Farbraum, Datum, GPS-Daten etc.). Für die Beschreibung des Bildinhalts ist aus TIFF 6.0 alleine das Feld **ImageDescription** (Tag ID: 10E) übernommen worden. In diesem Feld wird üblicherweise der Titel des Bildes abgespeichert. Die Länge des Attributwerts ist variabel, jedoch kann nur ASCII als Zeichenkodierung verwendet werden. Zur Verwaltung von Notizen bzw. Benutzerkommentaren zu Bildern wurden im Exif-Standard die Felder **MakerNote** und **UserComment** hinzugefügt. Beide Felder haben eine variable Länge und keine vorgegebene Zeichenkodierung. Das Feld **MakerNote** ist ausschließlich für Exif-Daten erzeugende Anwendungen vorgesehen, um beliebige Information zu vermerken. Es sollte jedoch nicht für andere Zwecke (z. B. Inhaltsbeschreibung durch Benutzer) missbraucht werden. Hierfür dient das Attribut **UserComment**, welches die Einschränkungen bzgl. der Zeichenkodierung von **ImageDescription** aufhebt.

3.3.1.2 IPTC-IIM

Das in London ansässige International Press Telecommunications Council (IPTC) ist ein Zusammenschluss der wichtigsten Nachrichtenagenturen, Verlage und Nachrichtenlieferanten der Welt. Von IPTC werden Standards zum Nachrichtenaustausch spezifiziert, die weltweit von nahezu jeder Nachrichtenorganisation verwendet werden. 1991 wurde mit IPTC-IIM ein Standard zum Austausch von Nachrichtentexten und verknüpften Bildern erstellt. Dabei wurden die IPTC-Header für Bilddateien eingeführt, die heute von verschiedenster Bildbearbeitungssoftware genutzt werden. Zur Beschreibung des Bildinhalts hält IPTC sowohl einen eigenen, auf die Nachrichtenindustrie zugeschnittenen Katalog mit definierten Werten als auch zwei Felder für manuelle Texteingaben bereit. Außer der standardisierten Kategorien (siehe Abschnitt 3.3.2) gibt es noch die Möglichkeit unter **Keywords** Schlüsselwörter anzugeben. Das hierfür vorgesehene Feld fasst jedoch nur 64 Bytes. Für eine längere textuelle Beschreibung kann das Feld **Caption/Abstract** verwendet werden, welches 2 000 Bytes zur Verfügung stellt.

Nachdem Adobe einige Felder aus IPTC-IIM für sein XMP-Format übernommen hat, haben Adobe und IPTC gemeinsam den IPTC-Core [IPT08] definiert, was im wesentlichen nur einer XML-Umsetzung von IPTC-IIM entspricht mit wenigen neuen Attributen. Relevant aus den neuen Attributen ist lediglich **Headline**, welches eine kurze Beschreibung des Bildinhalts ermöglicht. Das IPTC Extension Schema [IPT08] definiert weitere Felder zur Abspeicherung von Metadaten. Hier finden sich auch einige neue Attribute zur Beschreibung von Bildinhalten, welche im Anhang A in Tabelle A.1 aufgelistet sind.

3.3.1.3 XMP

In 2001 hat Adobe basierend auf verschiedenen bereits bestehenden Metadatenstandards (u. a. Dublin Core, IPTC, Exif) seinen neuen Standard XMP eingeführt. XMP basiert auf XML [XMP08a] und ermöglicht die Abspeicherung von Metadaten sowohl in der Mediendatei selber als auch in einer getrennten Datei [XMP08c]. Außerdem besteht die Möglichkeit eigene anwendungsspezifische Schemata in XMP einzubinden [XMP08b]. XMP standardisiert dadurch die Definition, Erstellung und Bearbeitung von erweiterbaren Metadaten für diverse Medienformate.

Ausgehend von der Erweiterbarkeit von XMP wurden von Adobe verschiedene allgemeine und spezialisierte Schemata definiert. Interessant aus der Sicht der inhaltsbe-

schreibenden Attribute sind die folgenden Schemata (in Klammern die entsprechenden Präfixe):

- Dublin Core Schema (*dc*)
- XMP Basic Schema (*xmp*)
- Photoshop Schema (*photoshop*)
- Exif Schema für TIFF-Eigenschaften (*tiff*)
- Exif Schema für Exif-spezifische Eigenschaften (*exif*)
- IPTC Core Schema (*IPTC 4xmpcore*)
- IPTC Extension Schema (*IPTC 4xmpext*)
- Publishing Requirements for Industry Standard Metadata (PRISM) Schema (*prism*)
- Digital Image Submission Criteria (DISC) Schema (*disc*)

Die entsprechenden bildbeschreibenden Attribute aus diesen Schemata sind im Anhang A in den Tabellen A.4 und A.5 aufgelistet. Die aus Exif bzw. TIFF übernommenen Attribute wurden in Abschnitt 3.3.1.1, die aus IPTC-Core und IPTC Extension adoptierten Felder wurden in Abschnitt 3.3.1.2 und Tabelle A.1 näher vorgestellt.

3.3.2 Konzept-Thesauri für Bildannotationen

Um die Annotation von grafischen Inhalten (Bilder und Videos) zu vereinheitlichen, wurden mehrere verschiedene Konzept-Thesauri von diversen Arbeitsgruppen vorgeschlagen. In diesem Abschnitt werden die wichtigsten Termsammlungen kurz vorgestellt.

Zur Anreicherung von Bildannotationen wird in vielen Ansätzen die WordNet Ontologie¹ [Mil95] verwendet. WordNet ist eine lexikalische Datenbank für englische Wörter, die speziell für die maschinelle Verwendung konzipiert wurde. Mittlerweile wurde WordNet durch die Global WordNet Association² in 50 Sprachen umgesetzt, wobei die einzelnen Sammlungen untereinander leider nicht verknüpft sind und somit auch keine Übersetzung der Konzepte möglich ist. In WordNet 3.0 sind nahezu alle Substantive, Verbe, Adjektive und Adverbien der englischen Sprache in Synonymmengen, sog. Synsets zusammengefasst, die alle jeweils ein Konzept repräsentieren. Die einzelnen Synonymmengen sind

1 <http://wordnetweb.princeton.edu/perl/webwn>

2 <http://www.globalwordnet.org/>

durch verschiedene semantische Beziehungen miteinander verbunden. Die semantischen Beziehungen zwischen den Wörtern sind in Anhang B näher beschrieben.

Im Thesaurus of Graphical Material (TGM-I¹) wurden 11 926 Terme zur Annotation von Bildern und Videos zusammengetragen. Alle Terme haben eine eindeutige ID und es sind auch die Obermengen bzw. die Spezialisierungen vermerkt.

IPTC hat in seinem Metadatenstandard zusätzlich einen Katalog für die Annotation von Bildern erstellt. Der Katalog beschreibt auf der höchsten Hierarchiestufe große Themengruppen, wie z. B. Politik, Bildung, Katastrophen und Unfälle, Gesundheit, Religion oder Sport. Auf zweiter Hierarchieebene werden diese Themengebiete weiter unterteilt, wie z. B. Bildung in die Unterkategorien Ausbildung, Fortbildung, Elternräte, Vorschulen, Schulen, Lehrervereinigungen und Universitäten. Jede dieser Kategorien ist durch eine eindeutige 8-stellige Zahl kodiert. Die Namen und Definitionen der Codes sind auf der Webseite von IPTC² in verschiedenen Sprachen abrufbar.

TV-Anytime³ hat in ihrer Metadata Specification Version 1.3⁴ insgesamt 954 Terme für die Kategorisierung von Fernsehinhälften zusammengestellt. Die einzelnen Terme sind ähnlich wie bei IPTC hierarchisch geordnet und durch IDs eindeutig identifizierbar.

In [NST⁺⁰⁶] wurde eine Konzeptsammlung für visuelle Medien vorgestellt. Der Large-Scale Concept Ontology for Multimedia (LSCOM) Standard beinhaltet 856 Terme, wovon 449 für den TREC Video Retrieval Evaluation (TRECVID) Wettbewerb 2006⁵ zur Annotation von Nachrichtenvideos eingesetzt wurden. Die Terme können, wie in [NST⁺⁰⁶] beschrieben, auch in eine Hierarchie eingeordnet werden, jedoch sieht der technische Bericht zum Standard in [Ken06] keine Hierarchie vor.

In [GHP07] wurde bei der Vorstellung des Caltech 256 Datensatzes auch eine hierarchische Taxonomie der verwendeten Begriffe erstellt. Auf der Webseite⁶ des Caltech Datensatzes ist auch eine weitere hierarchische Taxonomie verfügbar.

Escort ist eine Initiative der EBU zur Beschreibung und Kategorisierung von Fernsehinhälften. In Escort 2.4 [ESC07] wurden insgesamt 115 Terme zur Kategorisierung von Fernsehinhälften hierarchisch definiert.

1 <http://www.loc.gov/rr/print/tgm1/>

2 <http://cv.iptc.org/newsCodes/subjectCode/>

3 <http://www.tv-anytime.org>

4 ftp://tva:tv@ftp.bbc.co.uk/Specifications/COR3_SP003v13.zip

5 <http://www-nlpir.nist.gov/projects/tv2006/tv2006.html>

6 <http://www.vision.caltech.edu/CaltechChallenge2007/results/trees/tree256b.png>

Die zwei größten Anbieter von digitalen Bildern iStockphoto¹ und gettyimages² haben zur Erleichterung der Suche insgesamt 163 bzw. 300 Terme spezifiziert. Beide verwenden für die Strukturierung der Terme nur eine flache Hierarchie mit 2 Ebenen.

NAME	TERME	HIERARCHIE
Caltech 256	256	✓
Escort 2.4	115	✓
gettyimages	300	–
IPTC	1.405	✓
iStockphoto	163	–
LSCOM	856	–
TGM-I	11.926	✓
TV-Anytime	954	✓

Tabelle 3.5: Vergleich von verschiedenen Konzept-Thesauri für die Annotation von visuellen Medien.

3.3.3 Zusammenfassung

XMP bietet offenbar die vielfältigsten Möglichkeiten zur Beschreibung von Bildinhalten, da es durch die erweiterten Schemata u. a. Exif und IPTC subsummiert. Ein weiterer Vorteil ist, dass XMP durch eigene anwendungsspezifische Schemata frei erweitert werden kann. Offen bleibt jedoch, welche Metadaten von den Programmen letztendlich tatsächlich genutzt werden bzw. in welcher Form dies geschieht. Adobe Lightroom³ speichert z. B. die Kategorienhierarchie durch Punkte getrennt im Beschreibungsfeld. Andere Programme können das gleiche Feld zwar mit einer Volltextsuche durchsuchen, jedoch bleibt die Hierarchie unberücksichtigt. Das heißt, eine frei gestaltbare Kategorienhierarchie ist mit den vorhandenen Feldern nur über Umwege realisierbar und wird auch von verschiedenen Programmen unterschiedlich gehandhabt. Lässt man die Kategorienhierarchie außer Acht, können Objekte, Personen, Ereignisse und Orte bequem in vorhandene Felder abgespeichert werden.

Zur möglichst einheitlichen Annotation von Bildern wurden verschiedene Konzept-Thesauri erstellt. Am häufigsten wurde in wissenschaftlichen Veröffentlichungen zur Annotation von Bildern bislang WordNet verwendet. Der Einsatz weiterer Kategorienhier-

1 <http://www.istockphoto.com/>

2 <http://www.gettyimages.com/>

3 <http://www.adobe.com/products/photoshoplightroom/>

archien sowie die Überprüfung von deren Beitrag zur Verbesserung der Klassifikation von Objekten bzw. zur Annotation von Bildern steht zur Zeit noch aus.

3.4 Zusammenfassung

In diesem Kapitel wurden Bildannotationen aus verschiedenen Blickwinkeln untersucht. In Abschnitt 3.1 wurden zur Motivation zwei Gebiete identifiziert, die eine automatisierte textuelle Annotation von Bildern dringend benötigen: die textbasierte Suche nach Bildern und die Betrachtung von Bildern bei Menschen mit Sehbehinderungen. Die Anforderungen beider Gebiete wurden in Abschnitt 3.1.2 und Abschnitt 3.2 im Detail untersucht. Die Erkenntnis war, dass sowohl sehende als auch sehbehinderte Menschen vor allem Objekte und Personen im Bild textuell annotiert haben möchten. Abstrakte Konzepte sowie Eindrücke und Gefühle wurden als weniger wichtig eingestuft und müssen nicht annotiert werden. In Abschnitt 3.3 wurden Standards zur Abspeicherung von Bildmetadaten sowie Konzept-Thesauri zur Annotation von visuellen Inhalten kurz vorgestellt. Der etablierteste und am besten erweiterbare Standard für Bildmetadaten ist XMP.

KAPITEL 4

AUTOMATISCHE VERFAHREN ZUR BILDANNOTATION

Mit der Entwicklung von neuen Ansätzen und Algorithmen hat sich in den letzten Jahren viel auf dem Gebiet der Bildannotation bzw. des Bildtaggings getan. Viele Ideen stammen aus der Textdomäne, wo sich die Annotation mittels Schlüsselwörtern bereits etabliert hat. Als Grundlage dienen neue Erkenntnisse der Objekterkennung in Bildern, welche den Bildinhalt versuchen in Text zu übersetzen. Aktuelle Ansätze zur Objekterkennung werden in Abschnitt 4.1 behandelt.

Das Gebiet der Annotation kann auf unterschiedliche Arten strukturiert werden. Zum einen können nach dem Grad des manuellen Eingriffs manuelle, semi- und vollautomatische Annotationsverfahren unterschieden werden. Aus der Perspektive der Quelle des Wissens können die Ansätze aufgeteilt werden auf die Ermittlung von Beschreibungen basierend auf dem Bildinhalt (durch annotierte Trainingsbilder erlernt), auf dem Kontext (z. B. in einer Webseite) oder durch diverse Arten von Social Tagging bzw. manuelle Beschreibung des Inhalts. Aus algorithmischer Sicht können die Ansätze auch auf probabilistische und nicht probabilistische Verfahren aufgetrennt werden. In [DJLW08] werden die Ansätze der automatischen Bildannotation auf die gemeinsame Modellierung von Wörtern und Bildern sowie auf die teilweise unter manueller Beteiligung geführte

Kategorisierung aufgeteilt. Die Ansätze zur Bildannotation aus der Literatur werden in Abschnitt 4.2 nach dem Grad des manuellen Eingriffs gegliedert kurz vorgestellt.

Die Erkenntnisse aus den vorgestellten Grundlagen zur Objekterkennung und Annotationsverfahren werden in Abschnitt 4.3 zusammengefasst.

4.1 Objekterkennung in Bildern

Die allgemeine Objekterkennung in Bildern beschäftigt sich mit dem Problem Bilder aller möglichen Objekte (${}^{\rho}f(\mathbf{x})$) der Umwelt U (bzw. einer Teilmenge davon $\Omega \subset U$) den zugehörigen Klassen $\Omega_k \in \Omega$ korrekt zuzuweisen. Oft wird dabei das Problem auf Bilder eingeschränkt, welche nur einzelne Objekte darstellen. Bilder mit mehreren Objekten werden durch manuelle Markierungen, eine Abtastung mittels Suchfenster verschiedener Größen oder durch vollautomatische Segmentierung in Teilbilder zerlegt und jeweils die Objekte in den Teilbildern identifiziert. Die Klassen entsprechen im Wesentlichen eindeutigen Nummern, zu denen der Name des Objekts in Textform zugeordnet ist. Somit kann durch eine möglichst zutreffende Klassifikation eine textuelle Beschreibung für Bilder erstellt werden.

Die allgemeine Objekterkennung in Bildern ist äußerst komplex. In [JC02] und [Nie03] wurden die wesentlichen Schwierigkeiten zusammengefasst. Folgende Probleme müssen gelöst werden:

- Beleuchtung: Objekte erscheinen sehr unterschiedlich bei verschiedenen Lichtverhältnissen.
- Farbvariationen: Farben können beim gleichen Objekt unterschiedlich sein.
- Hintergrund: die Trennung vom Hintergrund bzw. von anderen Objekten ist schwierig.
- Oberfläche: matte, spiegelnde und transparente Materialien.
- Objekttypen: starre, verformbare, flexible und andere Objekte (z. B. Wolken, Wasser).
- Perspektive: je nach Gesichtspunkt kann ein Objekt völlig anders erscheinen.
- Skalierung: Objekte tauchen in Bildern in verschiedenen Größen auf.

- Stördaten: bislang nicht modellierte, dem System unbekannte Objekte können bei der Erkennung für Verwirrung sorgen.
- Störeffekte: Schatten oder Reflexionen erschweren die Erkennung.
- Transformation: das Objekt kann um beliebige Achsen rotiert werden.
- Verdeckung: Teile der Objekte sind durch andere Objekte im Bild verdeckt.
- Zusammenstellung: Einzelobjekte und Objektgruppierungen.

Im letzten Jahrzehnt sind unzählige Veröffentlichungen zur Erkennung von Objekten in Bildern erschienen, welche zahlreiche Lösungsideen für die obigen Schwierigkeiten präsentieren. Die meisten dieser Ansätze untersuchen das Problem der Objekterkennung hauptsächlich aus dem Blickwinkel der effizienten visuellen Repräsentation von Bildern. Hierbei werden unterschiedliche Merkmale, Distanzmaße und Bildbeschreibungsmodelle konzipiert und verglichen. Die Erstellung effizienter, skalierender und erweiterbarer Objekterkennungssysteme blieb dabei im Hintergrund. Dadurch wurde auch der reale Einsatz von diesen neuen Verfahren in Bilddatenbanken oder im Web bislang kaum diskutiert.

Im folgenden Abschnitt wird ein Überblick zu den aktuellen Verfahren der Objekterkennung in Bildern gegeben. Zuerst wird in Abschnitt 4.1.2 das zur Zeit erfolgreichste Vorgehen zur allgemeinen Objekterkennung, das Bag-of-Words-Konzept (Bag of Words (BoW)) im Detail vorgestellt. Danach werden in Abschnitt 4.1.3 die am häufigsten verwendeten Datensätze kurz beschrieben. Bewertungsgrundlagen zum Vergleich von Objekterkennungsansätzen werden in Abschnitt 4.1.4 behandelt. Anschließend werden in Abschnitt 4.1.5 die wichtigsten Wettbewerbe erläutert. In Abschnitt 4.1.6 werden die am häufigsten zitierten und besten Verfahren kurz vorgestellt. Abschnitt 4.1.7 gibt einen Überblick über erste Ansätze zur hierarchischen und skalierenden Objekterkennung in Bildern. Schließlich wird in Abschnitt 4.1.8 der aktuelle Stand der Forschung auf diesem Gebiet zusammengefasst.

4.1.1 Allgemeine Einordnung von Objekterkennungsverfahren

Vor allem zu der Entwicklung neuer Merkmale und der Klassifikation bzw. Erkennung von Objekten und Szenen gab es in den letzten 10 Jahren einen großen Zuwachs an Publikationen. Dabei bildeten sich unterschiedliche Lösungsstrategien aus.

In [BMM07] werden diese grob auf low-level und semantische Strategien (siehe Ab-

bildung 4.1) aufgeteilt. Zum Ersteren gehören Ansätze die global oder für eine feste Bildaufteilung Merkmale extrahieren. Basierend auf den erlernten Zuordnungen aus der Stichprobe wird anschließend versucht die Zugehörigkeit von neuen Bildern richtig abzuschätzen. Die semantischen Strategien werden auf die Erkennung von Objekten, Konzepten und Eigenschaften aufgeteilt.

In einer vergleichenden Untersuchung der verschiedenen Ansätze in [BMM07] und [BR07] stellten sich die auf dem BoW-Konzept basierenden Verfahren als die besten Klassifikationsmethoden heraus. Diese unabhängige Untersuchung bestätigte auch die Messungen in diversen anderen Veröffentlichungen, in denen jeweils ein selbst vorgestelltes Verfahren evaluiert wurde. Im folgenden wird das BoW-Konzept vorgestellt sowie die wichtigsten Ansätze beschrieben.

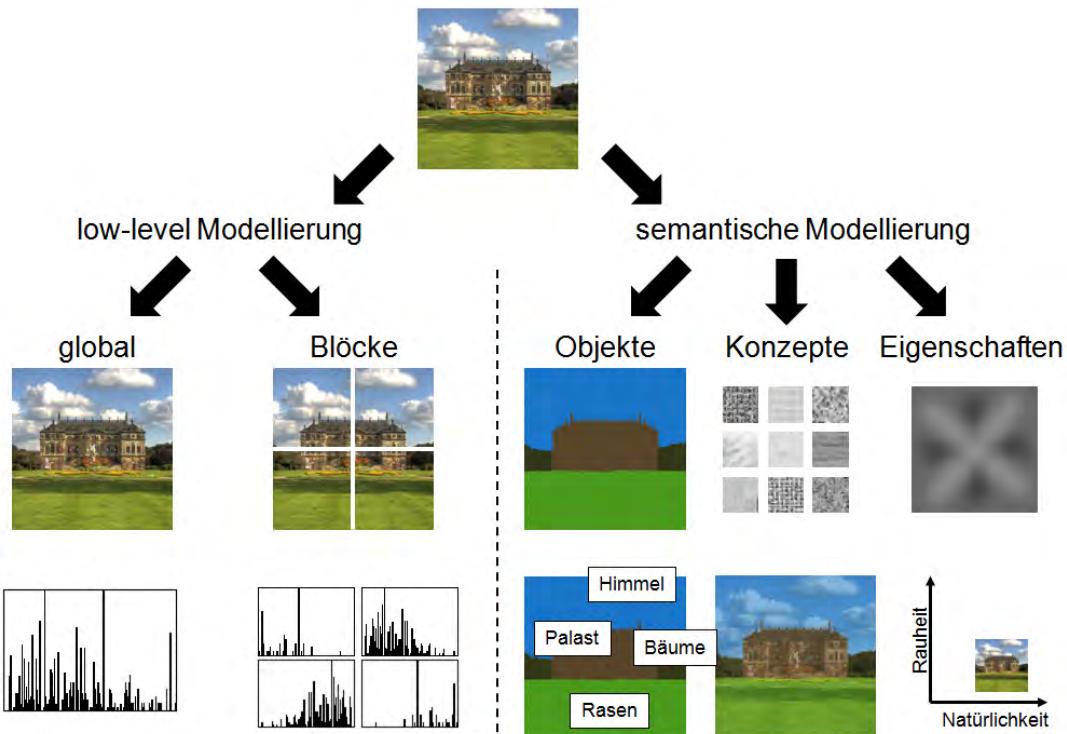


Bild 4.1: Strategien zur Klassifikation von Objekten und Szenen nach [BMM07].

4.1.2 Das Bag-of-Words-Konzept

Das BoW-Konzept stammt ursprünglich aus dem Text-Mining-Umfeld, die historischen Wurzeln gehen jedoch bis auf den Rosetta-Stein zurück [Bud29]. Auf diesem Stein ist ein und der selbe Text in drei verschiedenen Sprachen enthalten: griechisch, ägyptisch (Schreibform) und ägyptisch (Hieroglyphen). Erst durch den Fund dieses Steines konnte

die ägyptische Hieroglyphenschrift entschlüsselt werden. Allgemein angewendet werden im Text-Mining zwei Texte herangezogen, die den gleichen Inhalt und auch die gleiche Satzreihenfolge haben, jedoch in unterschiedlichen Sprachen verfasst wurden. Basierend auf dem Text-Mining der Sätze und Wörter kann ein Wörterbuch automatisch aus den Texten ermittelt werden. Dieses Wörterbuch kann im Anschluss zur Übersetzung von neuen Texten herangezogen werden. Eine ausführliche Beschreibung der mathematischen Grundlagen für diesen Text-Mining-Ansatz ist in [BDPDPM93] zu finden.

Der Ansatz kann auch auf Bilder übertragen werden, indem man interessante Punkte in Bildern sucht und diese mit geeigneten Methoden beschreibt. Die Übertragbarkeit des BoW-Ansatzes auf Bilder wurde zuerst in [MTO99] und [DBFF02] angesprochen. Um die Analogie zum Text-Mining-Verfahren herzustellen, werden die interessanten Punkte in Bildern auch *visuelle Wörter* genannt. Dabei sind unter dem Begriff „Punkte“ in diesem Sinne nicht nur einzelne Pixel, sondern auch deren direkte Umgebungen zu verstehen, weswegen sie auch oft als „Patches“ („Bildflecken“) bezeichnet werden. Die Menge der visuellen Wörter je Bild ergibt den „visuellen Text“, auch „Sack von Wörtern“ bzw. „Bag of Words“ genannt. Zusätzlich sind zu den Bildern der Stichprobe auch Texte annotiert, die den Inhalt des Bildes wiedergeben. Durch die Adaption des Text-Mining-Ansatzes auf visuelle Wörter und Textannotationen kann aufbauend auf eine Trainingsmenge ein Wörterbuch zwischen visuellen Bildinhalten und textuellen Bildbeschreibungen erstellt werden. Dieses Wörterbuch kann später für neue Bilder eingesetzt werden, um automatisch eine Textannotation zu ermitteln.

Die Verwendung von einzelnen interessanten Punkten als visuelle Wörter wird auch von den Ergebnissen in [Bie87] bzgl. der menschlichen Objekterkennung motiviert. In dieser psychologischen Untersuchung wurde die Erkennung von Objekten auf Teile der Objekte bzw. auf eine Menge von visuellen Primitiven zurückgeführt. Es wurden dabei auch Analogien zur Sprache und zur Bildung von Wörtern festgestellt.

Abbildung 4.2 zeigt das spezialisierte Vorgehen für das Erlernen von Klassen, Abbildung 4.3 das für die Annotation von Bildern mittels des BoW-Ansatzes nach dem Schema aus Abbildung 2.1. Die einzelnen Schritte werden in den nachfolgenden Abschnitten detailliert vorgestellt.

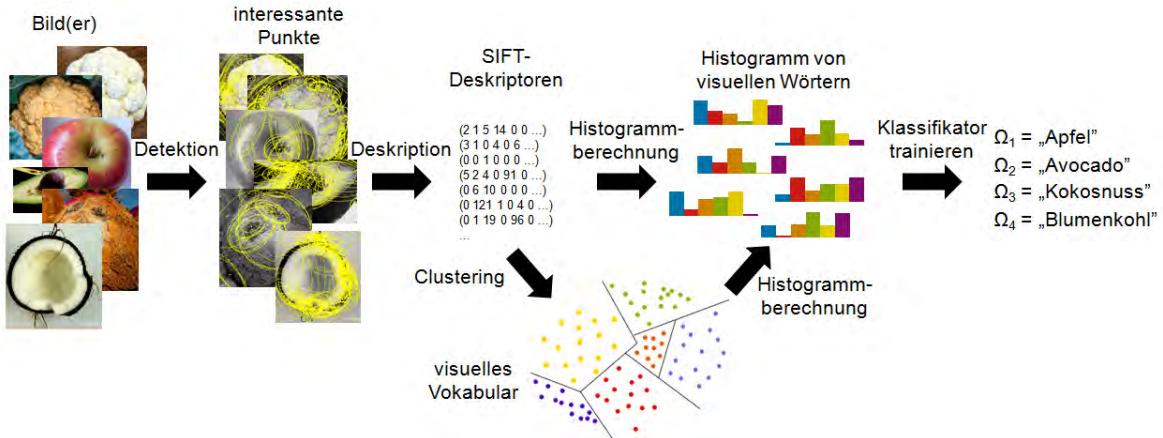


Bild 4.2: Grundsätzliches Vorgehen bei der Erlernung von Objektklassen mittels des Bag-of-Words-Ansatzes.

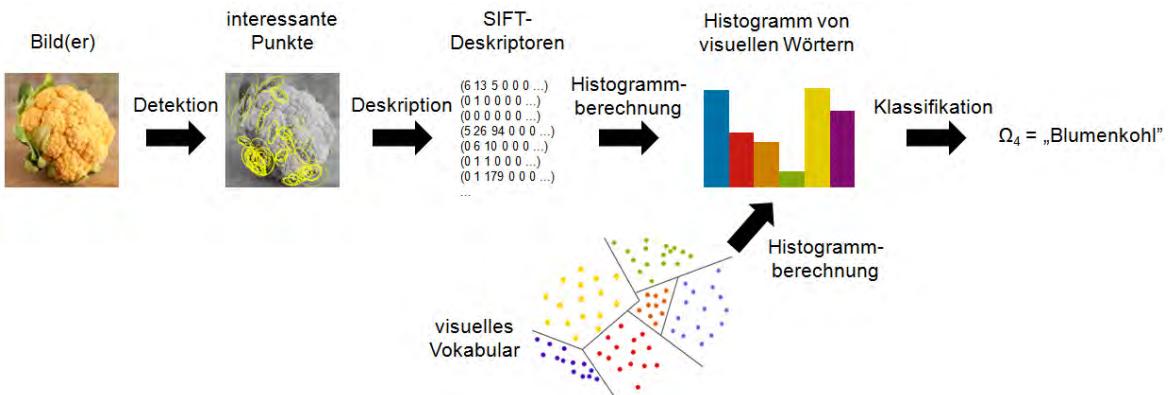


Bild 4.3: Grundsätzliches Vorgehen bei der Objekterkennung in Bildern mittels des Bag-of-Words-Ansatzes.

4.1.2.1 Detektoren

Beim BoW-Ansatz gibt es mehrere Einflussfaktoren, mit deren Veränderung bzw. Anpassung die Erkennung und die Zuordnung von Bildern zu Klassen verbessert werden kann. Zum einen existieren viele verschiedene Verfahren zur Ermittlung von interessanten Punkten, die je nach Einsatzgebiet mehr oder weniger geeignet sein können. Als Eingabe für diese Detektoren wird ein vorverarbeitetes Bild benötigt, d. h. das Bild muss bereits in den entsprechenden Farbraum (meistens Grauwerte) konvertiert und ggf. auf eine einheitliche Größe bzw. einheitliche Größenstufen skaliert worden sein. Letzteres ist deswegen nötig, da je nach Bildgröße und dem damit verbundenen Detaillierungsgrad der Bilder mit dem selben Detektor eine unterschiedliche Anzahl von interessanten Punkten gefunden wird.

Die Ermittlung von interessanten Punkten gleicht im Wesentlichen einer Abtastung der Bilder. Zur Bestimmung der interessanten Punkte kann entweder ein Gitternetz auf ein Bild gelegt (dense Sampling) oder es können Detektoren eingesetzt werden (sparse Sampling). [MSHW07] erläutert, dass auch eine Kombination der zwei Sampling-Methoden sinnvoll ist und zu Verbesserungen führen kann. Die Detektoren können von einer einfachen Kanten- bzw. Eckenerkennung bis zu komplexen zusammengesetzten Methoden variieren. Als Ergebnis erhält man eine Liste von Punkten, Ellipsen oder Regionen. Als „Blobs“ werden meistens ellipsenförmige Bildbereiche referenziert, während Regionen auch diverse andere Formen annehmen können.

[MTS⁺05] gibt einen Überblick und Vergleich über sog. affine Detektoren, d. h. Detektoren die invariant bezüglich Rotation, Skalierung, Helligkeitsänderungen und Ausschnitten sind. Für die Evaluation wurden Bilder mittels der genannten Transformationen verzerrt und anschließend die Wiederholbarkeit der Erkennung von interessanten Punkten sowie deren flächenmäßige Überlappung gemessen. Bei der Auswertung erwies sich Maximally Stable Extremal Region (MSER) für viele Transformationen als das beste Verfahren, dicht gefolgt vom Hessian-Affine-Detektor. In [TM08] sind die Definitionen und ein umfassender Vergleich von verschiedenen Detektoren zu finden.

Tabelle 4.1 stellt einen Vergleich zwischen gängigen Detektoren vor. In Abbildung 4.4 ist der Unterschied zwischen Ecken- und Blob-ähnlichen Detektoren anhand eines Beispielbildes visualisiert.

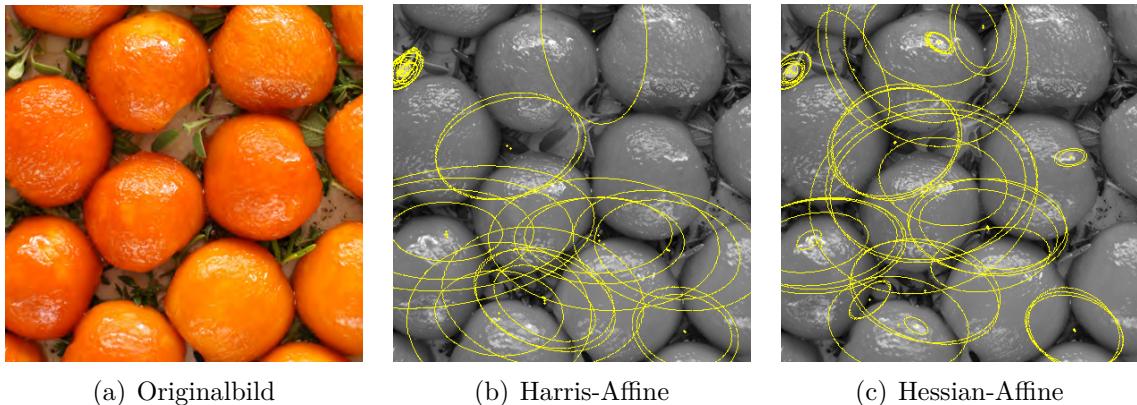


Bild 4.4: Vergleich von Ecken- (4.4(b)) und Blob-ähnlichen Detektoren (4.4(c)).

DETEKTOR	ROTATION	SKALIERUNG	AFFIN	ECKE	BLOB	REGION	BASIERT AUF
Difference of Gaussian (DoG)	✓	✓	–	(✓)	✓	–	approximiert Laplacian of Gaussian (LoG)
Harris	✓	–	–	✓	–	–	Matrix für Gradientenverteilung
Harris-Affine	✓	✓	✓	✓	(✓)	–	Harris-Laplace, affine Adaption [MS04]
Harris-Laplace	✓	✓	–	✓	–	–	Harris, LoG
Harris-Phase	✓	✓	–	✓	(✓)	–	Harris, [CJ03] und [Car04]
Hessian	✓	–	–	–	✓	–	Log
Hessian-Affine	✓	✓	✓	(✓)	✓	–	affine Adaption von Hessian-Laplace [MS04]
Hessian-Laplace	✓	✓	–	(✓)	✓	–	Determinante der Hessian-Matrix, LoG
hervorstechende Regionen (Kadir & Brady)	✓	✓	(✓)	(✓)	–	–	Wahrscheinlichkeitsdichtefunktionen von Intensitäten [MTS ⁺ 05]
intensitätsbasierte Regionen	✓	✓	✓	–	–	✓	Intensitätsebenen [TG04]
kantenbasierte Regionen	✓	–	–	✓	✓	✓	Harris, Curves, Canny, Berkeley
MSER	✓	✓	✓	✓	–	–	Schwellenwerte [MCMMP02]
SURF	✓	✓	–	(✓)	✓	–	approximierter Hessian
SUSAN	✓	–	–	✓	–	–	Intensitäten in der Umgebung von Punkten

Tabelle 4.1: Überblick über Detektoren für interessante Punkte in Bildern nach [MTS⁺05] und [TM08].

4.1.2.2 Deskriptoren

Ein weiterer Einflussfaktor bzgl. der Genauigkeit der Objekterkennung ist die Beschreibung des kompletten Bildes oder der interessanten Punkte mittels Merkmalen. Diese globalen oder lokalen Merkmale werden in der Objekterkennung oft als Deskriptoren bezeichnet. Sie können zur Ableitung von neuen Merkmalen oder direkt zum Training von Klassifikatoren sowie zum Erkennen von Objekten in neuen Bildern verwendet werden.

Für die Beschreibung der globalen Szene wird das komplette Bild als Eingabe benötigt. Hierfür hat sich der GIST-Ansatz (Spatial Envelopes) von [OT01, OT06] etabliert. GIST basiert auf Histogrammen von gerichteten Gradienten. Der Gradient bestimmt dabei die Richtung und den Betrag der größten (Grauwert-)Änderung innerhalb des Bildes bzw. Bildbereichs. Die Struktur der betrachteten Bilder wird in [OT01] durch die fünf Dimensionen Natürlichkeit, Offenheit, Rauigkeit, Unempfindlichkeit und Expansion beschrieben. Informationen über Objekte im Bild werden nicht berücksichtigt. GIST beschreibt Bilder in einer recht kompakten Form, deswegen wurde GIST auch von [DJS⁺09] für die Web-Bildsuche in Betracht gezogen, wobei es vor allem beim Auffinden von Duplikaten sehr gute Ergebnisse erzielt.

Bei den lokalen Merkmalen werden als Eingabe die Listen von Punkten bzw. Ellipsen und Regionen aus der Detektionsphase benötigt. Die Beschreibungen der von den Detektoren identifizierten interessanten Punkte müssen nach Möglichkeit ebenfalls invariant bzgl. Rotation, Skalierung und Helligkeitsänderungen sein. Die Deskriptoren können von einfachen Vektoren mit Pixelwerten bis zu komplexen Verfahren variieren. Meistens werden die Punkte nach Graustufen konvertiert (sofern das Bild in der Vorverarbeitung nicht schon bereit konvertiert wurde) und erst im Anschluss beschrieben. Es existieren jedoch auch Ansätze, welche die verschiedenen Farbkanäle (je nach Farbraum z. B. RGB oder HSV) jeweils einzeln nach Graustufen konvertieren und somit drei Beschreibungen für ein und denselben Punkt bzw. seiner Umgebung erhalten.

In [MS05] wurden die gängigsten Deskriptoren miteinander verglichen, inwieweit sie durch Rotation, Skalierung, Weichzeichnung, Belichtungswechsel, Veränderung der Perspektive und durch JPEG-Komprimierung beeinflusst werden. Zur Evaluation wurde die Anzahl der Übereinstimmungen zwischen Deskriptoren ermittelt aus Originalbildern und aus verzerrten Bildern herangezogen. Als beste Merkmale bzgl. der genannten Transformationen stellten sich diesbezüglich Gradient Location-Orientation Histogram (GLOH) und Scale-Invariant Feature Transform (SIFT) heraus. [SGS08] verglich verschiedene Farbdeskriptoren bzgl. ihrer Robustheit bei Helligkeitsunterschieden, Belichtungswechsel, Veränderung

der Lichtfarbe und Verschiebungen. Bei der Evaluation auf dem PASCAL VOC 2007 Datensatz (siehe Abschnitt 4.1.3) schnitten bzgl. der MAP die Deskriptoren W-SIFT und rgSIFT am besten ab. Basierend auf der Anzahl der gefundenen übereinstimmenden Punktpaare zwischen kameraperspektivisch variierten Bildern wurde in [MY09] der neue ASIFT-Ansatz für besser befunden als SIFT.

Eine Liste der unterschiedlichen Deskriptoren ist in Tabelle 4.2 zusammengefasst.

SIFT-Deskriptor

Zur Verdeutlichung, wie die Beschreibung eines Punktes funktioniert, wird der SIFT-Deskriptor nach [Low04] kurz vorgestellt.

Beim SIFT-Deskriptor werden entweder die von den Detektoren ermittelten Ellipsen auf Kreise normiert oder es wird die Umgebung (Fenster) von Punkten (Keypoints) für die weitere Berechnung herangezogen. Anschließend werden die Beträge $m(x,y)$ und Richtungen $\theta(x,y)$ der Gradienten in der Umgebung des Keypoints (x,y) (bzw. innerhalb des Kreises) unter Zuhilfenahme der Helligkeitswerte der Pixel ($L(x,y)$) mittels folgender Berechnungen ermittelt:

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \quad (4.1a)$$

$$\theta(x,y) = \arctan \frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)} \quad (4.1b)$$

Zur Erreichung der Rotationsinvarianz werden ein Orientierungs-Histogramm je 10° (insgesamt 36 Bins) anhand der Beträge und Richtungen der Gradienten erstellt. Anschließend wird das Fenster entsprechend der durch das Histogramm ermittelten dominanten Richtung gedreht.

Um große Veränderungen bei kleinen Verschiebungen des Fensters zu vermeiden und um das Zentrum eines Keypoints stärker zu berücksichtigen, wird das Fenster mit einer Gauß-Funktion gewichtet. Zuletzt werden die Fenster auf Regionen aufgeteilt und die Beträge bzw. Richtungen der Gradienten je Region zu einem Histogramm mit 8 Bins (je 45°) aufsummiert. Abbildung 4.5 stellt den SIFT-Deskriptor mit einem 8×8 Fenster und 2×2 Regionen vor, wobei der Kreis die Gauß-Gewichtung andeutet. Die Histogramme der einzelnen Regionen werden anschließend zu einem einzigen Histogramm konkateniert. Bei dem Beispiel in Abbildung 4.5 enthält der SIFT-Deskriptor (bzw. das konkatenierte Histogramm) 32 Dimensionen. In der Realität werden in allen Ansätzen 16×16 Fenster mit 4×4 Regionen eingesetzt, womit sich SIFT-Deskriptoren mit 128 Dimensionen ergeben.

DESKRIPTOR	QUELLE	BESCHREIBUNG
ASIFT	[MY09]	verbesserte SIFT-Variante mit erhöhter Robustheit bzgl. Drehungen und Veränderungen der Kameraachse
Differential-Invarianten	[KD87]	durch Faltungen berechnete Ableitungen der Gauß-Verteilung
Formkontext	[BMP02]	Histogramm von Kan tenpunkt positionen und Richtungen
GLOH	[MS05]	Erweiterung von SIFT mit erhöhter Robustheit und Unterscheidbarkeit
Histogramm (allg.)	–	Anzahl Pixelwerte je Pixelkategorie, Farbvarianten (siehe [SGS08]): RGB-Histogramm, Opponent Histogramm, Hue Histogramm, rgHistogramm, transformed color Histogramm
HoG	[DT05]	Histogramm von gerichteten Gradienten
komplexe Filter	[SZ02]	abgeleitet aus Gleichungen der Art $K(x,y,\theta) = f(x,y)\exp(i\theta)$
Kreuzkorrelation	[MS05]	Kreuzkorrelation als Vergleich zwischen geglätteten Regionen
Level Line Deskr.	[MSC ⁺ 06]	kompakte und affine Beschreibung als aussagekräftig eingestufter Kanten
Momente	[GMU96]	Momente zweiter Ordnung und zweiten Grades, Farbvarianten (siehe [SGS08]): Farbmomente, Farbmomentinvarianten
kAS und PAS	[FFJS08]	Skalierungsinvariante lokale Beschreibung von Formen repräsentiert durch Ketten von Kantenausschnitten mit k Verbindungen (PAS; $k = 2$)
PCA und DCT	[DHS00]	Reduktion auf Hauptkomponenten
PCA-SIFT	[KS04]	SIFT durch PCA reduziert
Pixelwerte	–	Vektor von Pixelwerten
RIFT	[LSP05]	Abwandlung von SIFT
SIFT	[Low04]	Histogramm von Gradienten und 8 verschiedenen Richtungen, Farbvarianten (siehe [SGS08]): HSV-SIFT, HueSIFT, OpponentsIFT, W-SIFT, rgSIFT, transformed color SIFT
Spin	[JH97, LSF05]	Histogramm von Distanzen und Intensitäten der umliegenden Punkte
steuerbare Filter	[FA91]	durch Faltungen berechnete Ableitungen der Gauß-Verteilung

Tabelle 4.2: Überblick über Deskriptoren für interessante Punkte in Bildern.

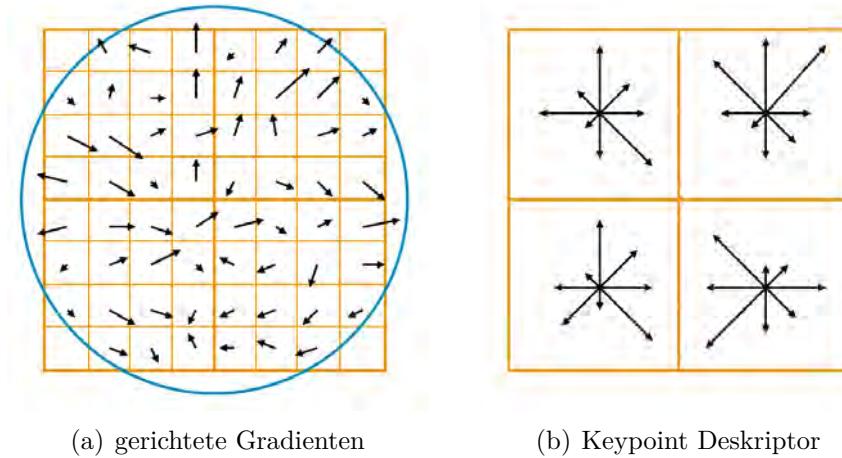


Bild 4.5: Beschreibung von Punkten mit dem SIFT-Deskriptor nach [Low04].

Listing 4.1 zeigt als Beispiel eine Liste von Punkten mit einem Ausschnitt der zugehörigen SIFT-Deskriptoren. Die erste Zeile beschreibt die Anzahl der Dimensionen je Punkt, in der zweiten Zeile steht die gesamte Anzahl der Punkte für das aktuelle Bild. Anschließend folgt zeilenweise die Liste der Punkte, wobei die ersten 12 Spalten die Position der Punkte bzw. die Größe derer Umgebung (Fenster) beschreiben.

4.1.2.3 Visuelles Vokabular

Da aus der Deskription der interessanten Punkte sehr viele unterschiedliche Beschreibungen resultieren, wird in nahezu allen Ansätzen in der Trainingsphase basierend auf einer zufällig bestimmten Untermenge der Punktbeschreibungen der Bilder der Stichprobe durch eine Art Quantisierung ein *visuelles Vokabular* erstellt. Als Algorithmus wird hierfür am häufigsten das k -means Clustering in Verbindung mit der euklidischen Distanz (oder andere Distanzmaße, siehe Abschnitt 4.1.2.5) eingesetzt, es kann aber auch z. B. ein

```

1 128
2 1564
3 10 10 1000 10 0 0 10 0 1 0 0 1 6 13 5 0 0 0 0 0 5 29 135 0 0 0 0 0 0 0 149 0 0 0 0 0 0 0 137 0 ...
4 10 10 1000 15 0 0 10 0 1 0 0 1 0 1 0 0 0 0 0 0 0 10 38 0 0 0 0 0 0 0 77 0 0 0 0 0 0 0 55 0 0 ...
5 10 10 1000 20 0 0 10 0 1 0 0 1 0 0 0 0 0 0 0 0 0 3 13 0 0 0 0 0 0 0 0 32 0 0 0 0 0 0 0 23 0 0 ...
6 10 10 1000 25 0 0 10 0 1 0 0 1 0 0 0 0 0 0 0 0 0 1 5 0 0 0 0 0 0 0 15 0 0 0 0 0 0 0 11 0 0 0 ...
7 20 10 1000 10 0 0 10 0 1 0 0 1 5 26 94 0 0 0 0 0 0 0 159 0 0 0 0 0 0 0 159 0 0 0 0 0 0 0 152 0 ...
8 20 10 1000 15 0 0 10 0 1 0 0 1 0 6 10 0 0 0 0 0 0 3 66 0 0 0 0 0 0 0 74 0 0 0 0 0 0 0 53 0 0 0 ...
9 20 10 1000 20 0 0 10 0 1 0 0 1 0 1 1 0 0 0 0 0 0 2 21 0 0 0 0 0 0 0 29 0 0 0 0 0 0 0 20 0 0 ...
10 20 10 1000 25 0 0 10 0 1 0 0 1 0 0 0 0 0 0 0 0 1 9 0 0 0 0 0 0 0 15 0 0 0 0 0 0 0 10 0 0 0 ...
11 30 10 1000 10 0 0 10 0 1 0 0 1 0 1 179 0 0 0 0 0 0 0 179 0 0 0 0 0 0 0 179 0 0 0 0 0 0 0 179 0 ...
12 ...

```

Listing 4.1: SIFT-Deskriptoren für mehrere Punkte.

Gaussian Mixture Model (GMM) verwendet werden wie in [GP08]. Als Ergebnis erhält man eine Menge von visuellen Wörtern, welche das allgemeingültige globale visuelle Vokabular bilden. Die Anzahl der visuellen Wörter im Vokabular variiert je nach der gewählten Anzahl k von Clustern. Das Vokabular wird in den meisten Ansätzen nur einmalig am Anfang bzw. während des Trainings berechnet und schränkt somit die Erweiterbarkeit der Systeme durch neue Objektklassen erheblich ein.

4.1.2.4 Histogramm von visuellen Wörtern

Nachdem aus den Bildern die interessanten Punkte durch Detektoren ermittelt und durch Deskriptoren beschrieben wurden, werden basierend auf den Deskriptoren und dem visuellen Vokabular für jedes Bild die Histogramme von visuellen Wörtern erstellt. Dabei werden die Punktdeskriptoren den nächsten Clusterzentren des visuellen Vokabulars zugeordnet und anschließend die Anzahl der Punkte je Cluster aufsummiert. Die Erstellung der Histogramme kann entweder für das gesamte Bild oder nur für Bildbereiche, welche durch Gitternetze (Grids) festgelegt werden, erfolgen. Diese Gitternetze sind nicht identisch mit dem Gitternetz für das dense Sampling bei der Detektion von interessanten Punkten. Die Bilder werden oft nur grob auf Bereiche aufgeteilt, wobei am häufigsten 3x1, 2x2 und 4x4 Aufteilungen verwendet werden. Die verschiedenen Gitternetze für Bildaufteilungen sind in Abbildung 4.6 dargestellt.

Durch die Kopplung verschiedener Detektoren mit unterschiedlichen Deskriptoren und Grids ergeben sich vielfältige Kombinationsmöglichkeiten, die in mehreren Ansätzen als „Kanäle“ bezeichnet werden. Sofern durch die Verwendung von Grids mehrere Histogramme für ein Bild entstehen, werden die einzelnen Histogramme konkateniert und zu einem einzigen Histogramm zusammengefügt. Das resultierende Histogramm kann durch geeignete Verfahren kompakter abgespeichert werden. [JDS09] stellt einige Möglichkeiten zur Komprimierung der BoW-basierten Beschreibungen von Bildern vor. Das Histogramm von visuellen Wörtern ist ein wesentliches Merkmal in der BoW-Kette, da es später zur Berechnung der Ähnlichkeit von zwei Bildern und zur Klassifikation verwendet wird.

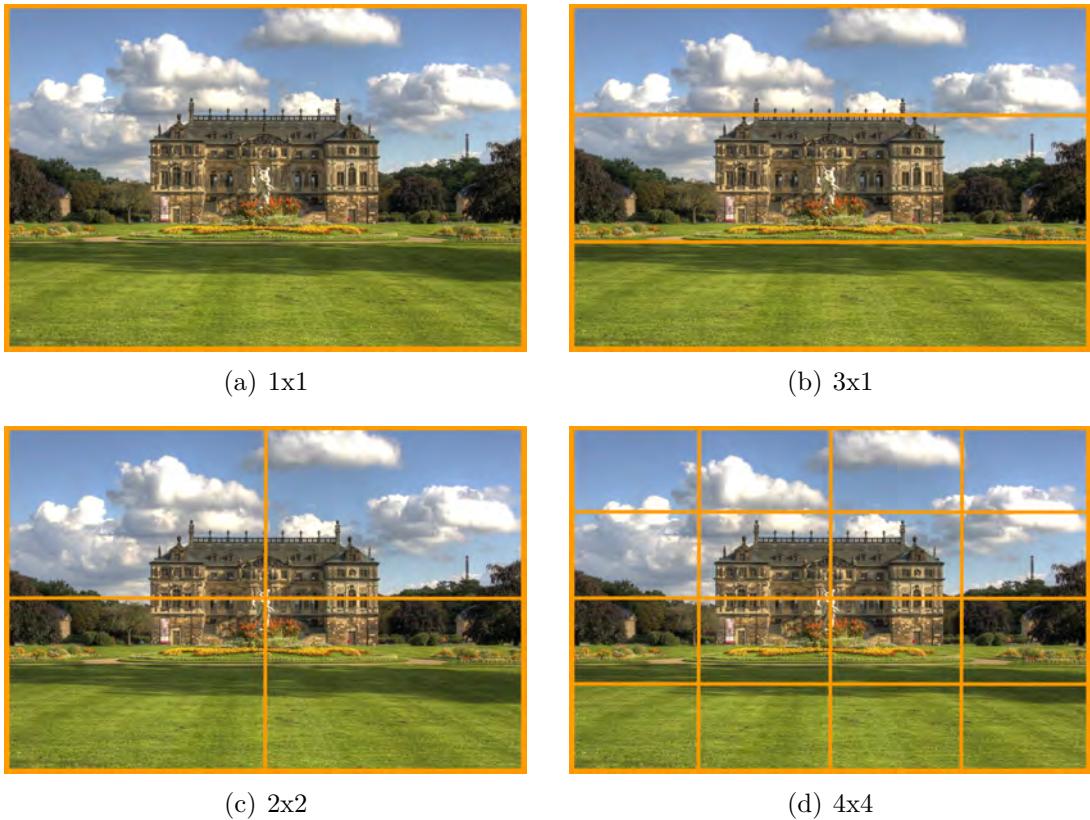


Bild 4.6: Beispieldurchsetzungen für Bilder mittels häufig eingesetzter Gitternetze.

4.1.2.5 Distanzmaße

Distanzmaße werden in der Objekterkennung bei zwei verschiedenen Schritten eingesetzt. Einerseits wird in den meisten Fällen die euklidische Distanz zur Erstellung des visuellen Vokabulars verwendet. Dabei werden die Abstände zwischen Punktbeschreibungen, wie z. B. SIFT-Deskriptoren, für das Clustering berechnet.

Andererseits werden Distanzmaße auch bei der Ermittlung des Klassifikators bzw. bei der Klassifikation von neuen Aufnahmen zur Berechnung der Ähnlichkeit zwischen zwei Bildern benötigt. Dabei werden Histogramme von visuellen Wörtern miteinander verglichen.

In [DJLW08] wird ein Überblick über die gängigsten Distanzmaße im Bildretrieval gegeben. Tabelle 4.3 listet diese und einige weitere Distanzmaße auf, wobei für jede Distanzfunktion vermerkt wird, ob es in den Objekterkennungsverfahren eher beim Clustering der Deskriptoren in visuelle Wörter (C) oder bei der Klassifikation von Bildern (K) eingesetzt wird.

DISTANMASS	DEFINITION	C / K
Cosinus	$\frac{F(x) \cdot G(x)}{\ F(x)\ \cdot \ G(x)\ }$	- / ✓
EMD	$\frac{\sum_{i=1}^m \sum_{j=1}^n f_{ij} d(p_i, q_j)}{\sum_{i=1}^m \sum_{j=1}^n f_{ij}}$	- / ✓
Euklidische Distanz (L2)	$(x - \mu)^t (x - \mu)$	✓ / ✓
Fidelity (Bhattacharyya)	$\sum_x \sqrt{F(x)} \sqrt{G(x)}$	- / ✓
Hamming Distanz	$h(b(x), b(y)) = \sum_{i=1}^{d_b} b_i(x) - b_i(y) $	✓ / -
Jensen Shannon (Jeffrey Divergenz)	$\sum_x F(x) \log \frac{2F(x)}{F(x)+G(x)} + G(x) \log \frac{2G(x)}{G(x)+F(x)}$	- / ✓
Kullback-Leibler Divergenz	$\sum_x F(x) \log \frac{F(x)}{G(x)}$	- / ✓
Mahalanobis	$(x - \mu)^t S^{-1} (x - \mu)$	✓ / ✓
χ^2	$\sum_x \frac{F(x) - G(x)}{F(x) + G(x)}$ oder $\frac{1}{2} \sum_i \frac{(u_i - w_i)^2}{u_i + w_i}$	- / ✓

Tabelle 4.3: Überblick über Distanzmaße in Objekterkennungsverfahren. Die rechte Spalte zeigt, ob das Distanzmaß bei Objekterkennungsverfahren hauptsächlich beim Clustering (C) oder bei der Klassifikation (K) zum Einsatz kommt. In der Earth-Mover Distanz (EMD) ist f_{ij} ein Gewichtungsfaktor und $d(p_i, q_j)$ steht für die Distanz zwischen den Clusterzentren p_i und q_j . $F(x)$ und $G(x)$ repräsentieren Dichten, typischerweise Histogramme. μ und x repräsentieren Vektoren, S kennzeichnet die Kovarianzmatrix.

4.1.2.6 Räumliche Beziehungen

Die Berücksichtigung von räumlichen Beziehungen zwischen visuellen Wörtern kann zu Verbesserungen in der Erkennung von Objekten führen. Dabei werden in den Objekterkennungsverfahren die räumlichen Beziehungen meistens erst nach der Erstellung des visuellen Vokabulars, d. h. erst nach der Quantisierung der durch Deskriptoren beschriebenen interessanten Punkte in visuelle Wörter untersucht. Somit wirkt sich die Position der interessanten Punkte innerhalb der Bilder nicht auf die Erstellung der visuellen Vokabulare aus.

[CL06] kategorisiert die verschiedenen Ansätze bezüglich der räumlichen Beziehungen zwischen visuellen Wörtern. Die Klassifikation der geometrischen Modelle ist in Abbildung 4.7 dargestellt. In erfolgreichen Verfahren, wie z. B. [GD05, GD07, MSHW07, BR07], wird häufig der Spatial-Pyramid-Ansatz nach [LSP06] zur Anreicherung der interessanten Punkte durch räumliche Beziehungen verwendet. Hierbei werden die Bilder hierarchisch

immer weiter durch Gitternetze unterteilt und für die jeweiligen Regionen Histogramme von visuellen Wörtern erstellt. Der Spatial-Pyramid-Ansatz wird in Abschnitt 4.1.2.7 im Detail vorgestellt.

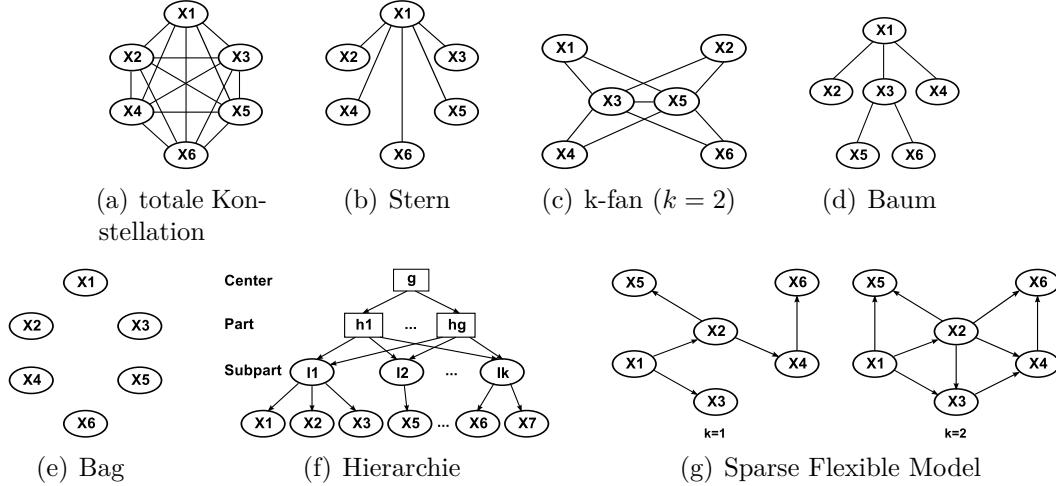


Bild 4.7: Klassifikation der Modelle zur Beschreibung der geometrischen Beziehungen zwischen Punkten nach [CL06].

4.1.2.7 Klassifikationsmethoden

Die Übersetzung von visuellen Inhalten in ihre textuelle Beschreibung ist im wesentlichen eine Klassifikation der Bilder in die zuvor erlernte Zuordnung von Kombinationen visueller Wörter zu Klassen. Da die Klassen Ω_κ mit Konzepten C_k einer Wissensbasis verknüpft sind, können somit für den Menschen verständliche Beschreibungen ${}^\rho\mathcal{B} = \langle I_j \langle C_k \rangle \rangle$ automatisch ermittelt werden (vgl. Abschnitt 2.2.2). Als Stichprobe ω für das Training eines Klassifikators $f \Rightarrow \Omega_\kappa$ werden die aus den Bildern ${}^\rho f(\mathbf{x})$ extrahierten Histogramme von visuellen Wörtern (Merkmale c), die zugeordneten Klassen (y_ρ) und evtl. a priori Wissen (z. B. in Form von Worthierarchien, Taxonomien o. Ä., siehe Abschnitt 4.1.7) benötigt. Als Ergebnis erhält man eine Art „Wörterbuch“, welches visuelle Inhalte ${}^\rho f(\mathbf{x})$ auf textuelle Konzepte ${}^\rho\mathcal{B}$ übersetzt. Bei der Klassifikation von neuen Bildern werden nur das Histogramm des Bildes und der erlernte Klassifikator benötigt. Letzteres liefert je nach Art eine oder mehrere Klassen Ω_κ bzw. die den Klassen zugeordneten Konzeptnamen C_k zurück.

Die Klassifikationsmethoden können nach [DJLW08] auf diskriminative und generative Verfahren aufgeteilt werden.

Bei diskriminativen Methoden werden die Klassengrenzen bzw. die zugehörigen a posteriori Wahrscheinlichkeiten direkt ermittelt. Formal bedeutet es, dass für neue Daten \mathbf{x} (bzw. Muster ${}^{\rho}\mathbf{f}(\mathbf{x})$) eine durch Trainingsdaten ω erlernte Funktion $g(\mathbf{x})$ ausgewertet wird, um die zugehörige Klasse Ω_{κ} bzw. die Wahrscheinlichkeit $p(\Omega_{\kappa}|\mathbf{x})$, dass das Objekt \mathbf{x} zur Klasse Ω_{κ} gehört, zu bestimmen. Beispiele für diskriminative Verfahren sind u. a. nächste Nachbarn (NN) Algorithmen, neuronale Netze, SVMs, Entscheidungsbäume oder Conditional Random Fields (CRF).

Generative Verfahren bestimmen die Dichte der Daten in jeder Klasse und wenden anschließend den Satz von Bayes zur Bestimmung der a posteriori Wahrscheinlichkeiten $p(\Omega_{\kappa}|\mathbf{x})$ an. Formal ausgedrückt bedeutet es, dass die Wahrscheinlichkeitsdichten für $p(\Omega_{\kappa})$ und $p(\mathbf{x}|\Omega_{\kappa})$ abgeschätzt werden und anschließend durch Anwendung des Satzes von Bayes ($p(\Omega_{\kappa}|\mathbf{x}) = \frac{p(\mathbf{x}|\Omega_{\kappa})p(\Omega_{\kappa})}{p(\mathbf{x})}$) $p(\Omega_{\kappa}|\mathbf{x})$ ermittelt wird. Beispiele für generative Verfahren sind u. a. Mischverteilungen, Latent Dirichlet Allocation (LDA), Hierarchical Dirichlet Process (HDP) oder Probabilistic Latent Semantic Analysis (pLSA).

Diskriminative und generative Objekterkennungsmethoden wurden bereits in mehreren Veröffentlichungen verglichen ([UB05, UB06, Hol07, PD07, BZM08]). Bei diskriminativen Verfahren können die Klassengrenzen einfacher optimiert werden, während bei generativen Verfahren a priori Wissen besser integriert werden kann. Generative Verfahren benötigen weniger Trainingsbeispiele als diskriminative Methoden und können nach [BZM08] in Kombination mit diskriminativen Verfahren bei steigender Anzahl von Klassen die Objekterkennung leicht verbessern. Diskriminative Verfahren sind bzgl. der Klassifikation bei einer genügend großen Stichprobe genauer und können zugehörige Klassen wesentlich schneller bestimmen. Nach [UB05] sind diskriminative Verfahren beim Training bis zu 20-fach und bei der Klassifikation bis zu 200-fach schneller als generative Verfahren. Zwar sind sich alle oben genannten Untersuchungen einig darüber, dass diskriminative und generative Verfahren nach Möglichkeit in Kombination verwendet werden sollten, jedoch ist der Gewinn an Genauigkeit in der Klassifikation gering und kann nur durch stark erhöhte Trainings- und Klassifikationszeit erreicht werden. Auch in aktuellen Wettbewerben sind vornehmlich diskriminative Ansätze am erfolgreichsten (vgl. Abschnitt 4.1.5 und 4.1.6). Aus diesen Gründen werden in dieser Arbeit allein diskriminative Verfahren betrachtet.

Bei der allgemeinen Objekterkennung in Bildern werden in vielen Ansätzen Support Vektor Maschinen (Support Vector Machine (SVM)) mit verschiedenen spezialisierten

Kernfunktionen eingesetzt. Im folgenden Abschnitt wird die Funktionsweise von SVMs sowie die wichtigsten Kernfunktionen in der Objekterkennung kurz vorgestellt.

Support Vektor Maschinen

Das grundlegende Konzept von SVMs wurde in [Vap95] und [SBV95] vorgestellt. SVMs sind in ihrer ursprünglichen Form nur zur binären Klassifikation geeignet. Möglichkeiten zur Erweiterung der Klassifikation auf mehrere Klassen werden in Abschnitt 4.1.7 beschrieben.

Die Grundidee eines SVMs basiert darauf, dass ausgehend von einer Stichprobe ω im Training eine Trenngerade (bzw. Hyperfläche) zwischen den Merkmalsvektoren von zwei Klassen Ω_k und Ω_v gesucht wird, welche das Risiko ein neues Muster (bzw. Merkmal) falsch zu klassifizieren minimiert. Sofern die Klassen Ω_k und Ω_v linear trennbar sind, wird aus der Vielzahl der möglichen Trenngeraden diejenige ausgewählt, welche den Abstand zwischen den nächstgelegenen Merkmalsvektoren der Klassen maximiert. Diejenigen Merkmalsvektoren, welche den gleichen Abstand zur Trenngerade besitzen, werden dabei als Support Vektoren bezeichnet. Abbildung 4.8 veranschaulicht linear trennbare Klassen, sowie die Trenngerade mit maximalen Abstand von beiden Klassen.

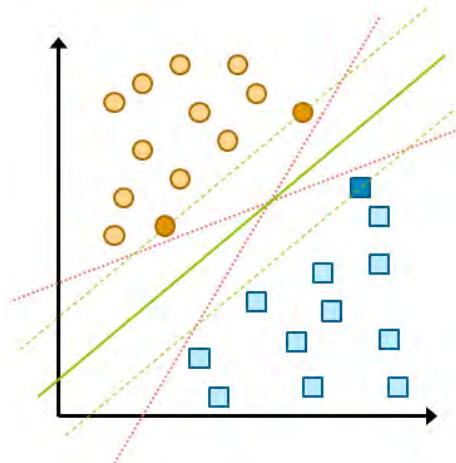


Bild 4.8: Linear trennbare Klassen mit möglichen (rot) und optimaler Trenngerade (grün). Die dunkel gefüllten Punkte (Merkmalsvektoren) entsprechen den Support Vektoren, da sie den gleichen Abstand zur optimalen Trenngeraden besitzen.

In der Regel sind die Klassen Ω_k und Ω_v linear nicht trennbar. Hierbei werden die Merkmalsvektoren in einen höherdimensionalen Merkmalsraum abgebildet und es wird versucht in diesem neuen Merkmalsraum eine Trennfläche zu finden. Abbildung 4.9 zeigt ein Beispiel für linear nicht trennbare Klassen in \mathbb{R}^2 sowie deren Abbildung mittels

$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}x_1x_2, x_2^2)$ in \mathbb{R}^3 und der entsprechenden Trennfläche nach [MMR⁺01].

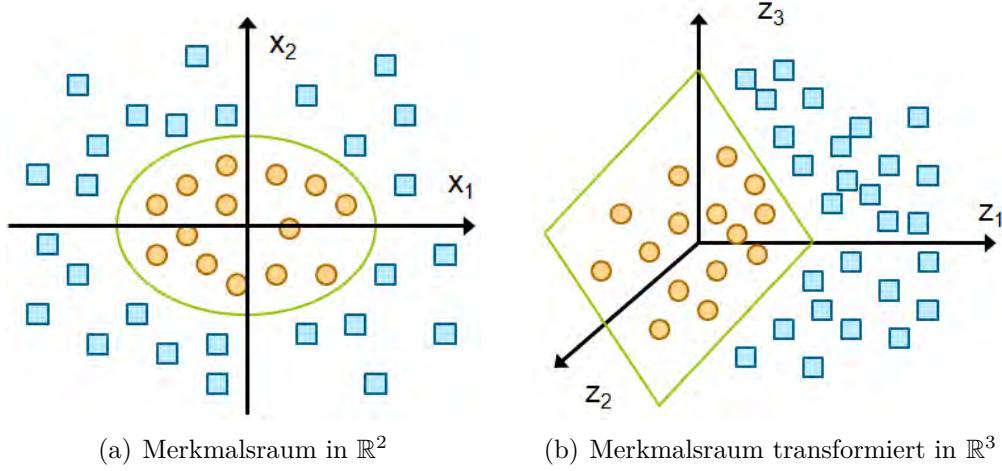


Bild 4.9: Linear nicht trennbare Klassen in \mathbb{R}^2 , deren Abbildung in \mathbb{R}^3 und die optimale Trennfläche nach [MMR⁺01]. Die Abbildung von \mathbb{R}^2 nach \mathbb{R}^3 erfolgte durch $(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}x_1x_2, x_2^2)$.

Nach der Transformation in einen höherdimensionalen Merkmalsraum $\Phi : \mathbb{R}^N \rightarrow \mathbb{R}^M$ müssen die Kreuzprodukte der Merkmalsvektoren im neuen Merkmalsraum \mathbb{R}^M berechnet werden. Da für die Bestimmung der optimalen Trennfläche nur diese Kreuzprodukte von Bedeutung sind, ist es durch den sog. „Kernel Trick“ [CV95] möglich auch ohne die Kenntnis von Φ und \mathbb{R}^M direkt aus den Merkmalsvektoren \mathbf{x} und \mathbf{y} das Kreuzprodukt zu berechnen. Dazu muss die sog. Kernfunktion $K(\mathbf{x}, \mathbf{y})$ die Bedingung von Mercer [Mer09, Vap95] erfüllen, welche sicherstellt, dass $K(\mathbf{x}, \mathbf{y})$ ein Kreuzprodukt in einem Merkmalsraum \mathbb{R}^M ist.

Die Bestimmung einer optimalen Kernfunktion ist auch heute noch ein ungelöstes Problem. In der Objekterkennung haben sich einige spezielle Kernfunktionen etabliert, welche nachfolgend kurz vorgestellt werden.

Kernfunktionen

Im Rahmen der allgemeinen Objekterkennung werden zur Klassifikation häufig SVMs eingesetzt (vgl. Abschnitt 4.1.6). Dabei wird mittels der nichtlinearen Abbildung der Merkmalsvektoren in einen neuen Merkmalsraum eine möglichst optimale Trennung von Klassen angestrebt. Die Abbildung und die Berechnung der Kreuzprodukte im neuen

Merkmalsraum wird durch Kernfunktionen abgekürzt. In diesem Abschnitt werden die in der Objekterkennung am häufigsten eingesetzten Kernfunktionen vorgestellt.

Einer der ersten speziell für die Objekterkennung vorgeschlagenen Kernfunktionen ist die Pyramid-Match-Kernfunktion von [GD05]. Die Idee bei diesem Ansatz ist es auf den Merkmalsraum immer feinere Gitter zu legen. Dabei verdoppelt sich die Anzahl der Zellen in den Gittern von einer einzigen Zelle bis hin zu einer feinen Aufteilung, bei der jeder Merkmalsvektor in eine eigene Zelle fällt. Insgesamt entstehen somit Gitter auf L Ebenen. Je Gitter und Bild wird ein Histogramm der Merkmalsvektoren erstellt, wodurch eine Kette von Histogrammen mit unterschiedlichen Auflösungen entsteht. Beim Vergleich der Merkmalsvektoren von Bildern X und Y werden auf jeder Ebene ℓ die Übereinstimmungen der Histogramme H_X^ℓ und H_Y^ℓ nach [SB91] folgendermaßen berechnet:

$$\mathcal{I}(H_X^\ell, H_Y^\ell) = \sum_{i=1}^{2^\ell} \min(H_X^\ell(i), H_Y^\ell(i)), \quad (4.2)$$

wobei 2^ℓ die Anzahl der Bins der Histogramme auf Ebene ℓ bezeichnet und $H_X^\ell(i)$ bzw. $H_Y^\ell(i)$ die Anzahl der Merkmale angibt, die sich auf der Ebene ℓ im Bin i befinden. Zuordnungen auf Ebene $\ell+1$ kommen automatisch auch auf der höheren Ebene ℓ vor. Um diese nicht doppelt zu berücksichtigen, entspricht die Übereinstimmung der Histogramme auf Ebenen $\ell = 0, \dots, L-1$ somit

$$\mathcal{N}(H_X^\ell, H_Y^\ell) = \mathcal{I}(H_X^\ell, H_Y^\ell) - \mathcal{I}(H_X^{\ell+1}, H_Y^{\ell+1}). \quad (4.3)$$

Zuletzt werden die Histogrammübereinstimmungen \mathcal{I} für alle Ebenen gewichtet aufsummiert. Diese Distanz zwischen zwei Bildern X und Y wird als Pyramid-Match-Kernfunktion wie folgt definiert:

$$K_\Delta(X, Y) = \sum_{\ell=0}^L w_\ell \mathcal{N}(H_X^\ell, H_Y^\ell), \quad (4.4)$$

wobei für das Gewicht w_ℓ in [GD05] $1/2^\ell$ vorgeschlagen wird.

In der allgemeinen Objekterkennung wird am häufigsten die Spatial-Pyramid-Kernfunktion von [LSP06] eingesetzt. Diese adaptiert die Pyramid-Match-Kernfunktion von [GD05], wobei in [LSP06] nicht der Merkmalsraum, sondern das Bild selber horizontal und vertikal stufenweise mittels immer feineren Gitternetzen aufgeteilt wird (vgl.

„Kanäle“ in Abschnitt 4.1.2.4). Für jede Zelle des Gitternetzes der Ebene ℓ wird ein Histogramm von visuellen Wörtern erstellt, womit auch der Pyramid-Match-Ansatz zusätzlich durch die Quantisierung der Deskriptoren mittels des visuellen Vokabulars ergänzt wird. Der Vorteil des Spatial-Pyramid-Ansatzes ist, dass auch die räumliche Position der visuellen Wörter innerhalb der Bilder mit berücksichtigt wird, was zu einer leichten Verbesserung in der Erkennung von Objekten führt. Abbildung 4.10 veranschaulicht die Spatial-Pyramid-Methode anhand eines Beispielbildes.

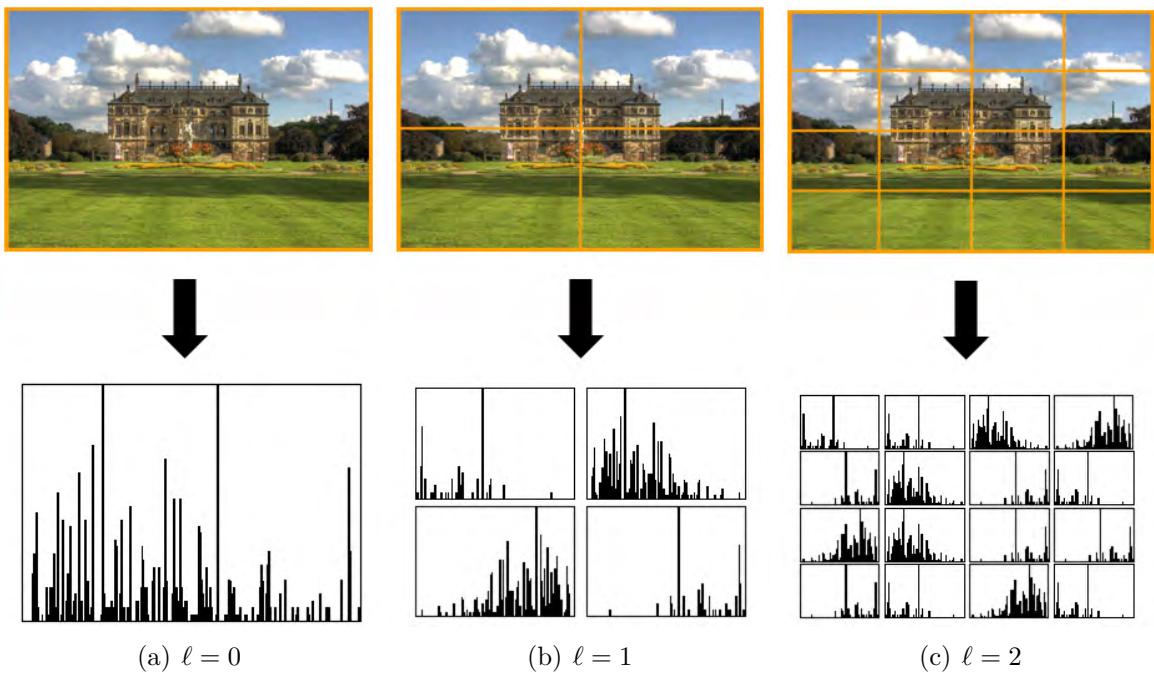


Bild 4.10: Beispiel für die Spatial-Pyramid-Methode nach [BR07].

Entsprechend den Änderungen bzgl. des Pyramid-Match-Ansatzes wird die Spatial-Pyramid-Kernfunktion nach [LSP06, BR07] folgendermaßen definiert:

$$K(X, Y) = \sum_{v=1}^V k'_\Delta(X_v, Y_v), \quad (4.5)$$

wobei V für die Anzahl der visuellen Wörter (bzw. für die Größe des Vokabulars) steht, $k'_\Delta(X_v, Y_v) = \sum_{\ell=0}^L w_\ell \mathcal{N}(H_{X_v}^\ell, H_{Y_v}^\ell)$ mit \mathcal{N} entsprechend der Formel 4.3 definiert ist, X_v und Y_v Mengen von zweidimensionalen Vektoren mit den Positionsangaben des visuellen Wortes v sind und L die Anzahl der Ebenen angibt. Üblicherweise werden Übereinstimmungen bei feineren Auflösungen höher gewichtet als Übereinstimmungen bei größeren Auflösungen. In [LSP06] wird eine ähnliche Gewichtung wie in [GD05]

verwendet, so dass die Gewichte w_ℓ auf $1/2^\ell$ gesetzt werden. Meistens wird die Spatial-Pyramid-Methode bis zur Ebene $L = 2$ eingesetzt.

In [ZMLS07] wurden erweiterte Gauß-Kernfunktionen eingesetzt, welche die Verwendung von beliebigen Distanzfunktionen ermöglichen. Gleichung 4.6 zeigt die Definition der erweiterten Gauß-Kernfunktion, wobei β ein durch Kreuzvalidierung ermittelbarer Skalierungsfaktor und $D(X,Y)$ eine beliebige Distanzfunktion zwischen zwei Histogrammen X und Y ist.

$$K(X,Y) = \exp\left(-\frac{1}{\beta}D(X,Y)\right) \quad (4.6)$$

In [BR07] wurde die Spatial-Pyramid-Kernfunktion von [LSP06] und die erweiterte Gauß-Kernfunktion aus [ZMLS07] in die Pyramid Radial Basis Function (P-RBF) Kernfunktion vereinheitlicht, welche wie in Gleichung 4.7 definiert ist.

$$K(X,Y) = \exp\left(\frac{1}{\beta} \sum_{\ell=0}^L w_\ell D_\ell(X,Y)\right) \quad (4.7)$$

Bei dieser Kernfunktion kann eine beliebige Distanzfunktion D_ℓ verwendet werden. β ist analog, wie in Gleichung 4.6 ein Skalierungsfaktor und w_ℓ ein von der jeweiligen Ebene ℓ abhängiges Gewicht.

4.1.2.8 Übergang zur Annotation

In den meisten Objekterkennungsansätzen wird lediglich die Klassifikation von Bildern ${}^{\rho}\mathbf{f}(\mathbf{x})$ in ihre zugehörigen Klassen Ω_κ behandelt. Für eine aussagekräftige textuelle Annotation \mathcal{B} ist die Verknüpfung von Klassen Ω_κ zu Konzepten C_k einer Wissensbasis nötig. Oft wird einfach der Name der zugeordneten Klasse in textueller Form als Annotation verwendet, einige Ansätze setzen auch Taxonomien oder Ontologien, wie z. B. WordNet ein. Die Konzepte C_k können als textuelle Repräsentation zu den Bildern annotiert werden, wobei je nach verwendeter Wissensbasis auch die Möglichkeit besteht durch Synonyme oder Hyperonyme die Annotation zu erweitern. Sofern ein Bild auf mehrere Regionen aufgeteilt wurde, kann ein Bild in seiner Gesamtheit durch die Klassifikation auch mehreren Klassen zugewiesen werden ($\mathbf{f} \Rightarrow \boldsymbol{\Omega} = [{}^1\Omega, \dots, {}^N\Omega]$), wobei dann alle zugehörigen Konzepte C_1, \dots, C_n zum Bild annotiert werden. Ansätze zur manuellen, semi- bzw. vollautomatischen Annotation von Bildern werden in Abschnitt 4.2 behandelt.

Im Folgenden werden die zur Evaluation von Objekterkennungsansätzen verwendeten Datensätze, Bewertungsmaße und Wettbewerbe vorgestellt.

4.1.3 Datensätze

Um die neu entwickelten Ansätze zu testen und mit bereits existierenden Methoden zu vergleichen, wurden viele verschiedene Datensätze erstellt. In diesem Abschnitt werden die gängigsten Datensätze zur Objekterkennung in Bildern kurz vorgestellt.

Der auf diesem Forschungsgebiet am meisten verwendete Datensatz ist Caltech 101¹ [FFFP04, FFFP06] bzw. Caltech 256² [GHP07] vom California Institute of Technology. Die Bilder wurden anhand der Google Bildsuche³ und PicSearch⁴ zusammengestellt. An dem Caltech 101 Datensatz wurde oft bemängelt, dass der Datensatz zu künstlich ausgelegt sei, da die Objekte in den meisten Bildern wenig Rotation aufweisen, zentral ausgerichtet und aus dem gleichen Blickwinkel abgelichtet sind. In Caltech 256 wurde der Datensatz unter Berücksichtigung dieser Kritiken erweitert.

Ähnlich populär zur Evaluation von Objekterkennungsverfahren sind die Datensätze der Pattern Analysis, Statistical modelling, Computational Learning (PASCAL) Wettbewerbe. Das PASCAL-Netzwerk⁵ veranstaltet seit 2005 jährlich verschiedene Wettbewerbe, unter anderem auch den Wettbewerb PASCAL Visual Object Class Challenge (PASCAL VOC) zur Objekterkennung, Segmentierung und zur Erkennung von menschlichen Gliedmaßen in Bildern. Der Datensatz wird jedes Jahr erweitert, die Anzahl der Klassen steht seit 2007 auf 20. In der Forschung wird am häufigsten der PASCAL VOC 2007 Datensatz verwendet, da nur für diesen Datensatz sowohl die Trainings- und Validierungs- als auch die Testbilder öffentlich zugänglich sind. Weitere Informationen zum PASCAL-Wettbewerb sind in Abschnitt 4.1.5 zu finden.

Der Corel⁶-Datensatz besteht aus 800 Photo-CDs zu verschiedenen Themengebieten mit jeweils 100 Bildern. Dieser Datensatz wurde sehr oft im Bereich des Bildretrievals eingesetzt, jedoch konnten die verschiedenen Forschungsergebnisse nur schlecht verifiziert

1 http://www.vision.caltech.edu/Image_Datasets/Caltech101/

2 http://www.vision.caltech.edu/Image_Datasets/Caltech256/

3 <http://images.google.com>

4 <http://www.picsearch.com>

5 <http://www.pascal-network.org>

6 <http://www.corel.com>

und miteinander verglichen werden, da jedes Projekt seine eigene Bilddatenbank aus dem kompletten Corel-Datensatz abgeleitet hat. Eine ausführliche Evaluation des Corel-Datensatzes ist in [MMMP02] zu finden.

In [TMF04] wurde der MIT CSAIL¹-Datensatz verwendet und zu Forschungszwecken bereitgestellt. Da er nur relativ wenige annotierte Bilder enthielt, wurde das Projekt LabelMe² ins Leben gerufen. Im LabelMe-Projekt können Benutzer vorhandene Bilder bearbeiten oder eigene Bilder hochladen und mit einem webbasierten Tool Polygone und Annotationen zu den Bildern hinzufügen. Die mit Polygonen versehenen und annotierten Bilder werden auch als Datensatz für die Forschung bereitgestellt. Eine detaillierte Beschreibung des Datensatzes ist in [RTMF08] zu finden.

Beim MIR Flickr³ [HL08] Datensatz wurde das Ziel verfolgt, eine Datenbank zu erstellen mit echten Bildern und mit echten Annotationen aus dem realen Umfeld. Hierfür wurden 25 000 Bilder von Flickr⁴ zu 30 vordefinierten Kategorien automatisch runtergeladen. Diese Datenbank wurde auch beim ImageCLEF 2009 Wettbewerb zur Annotation von Fotos verwendet. In 2010 wurde der Datensatz auf insgesamt 1 Million Bilder erweitert [HTL10]. Beide MIR Flickr Datensätze sind im wesentlichen nur Abzüge aus Flickr ohne jegliche weitere Bereinigung der Tags. Manche Bilder im Datensatz enthalten sogar keinerlei Annotationen.

In [DDS⁺09] wurde eine neue Bilddatenbank basierend auf der WordNet-Hierarchie vorgestellt. Das Ziel von ImageNet⁵ ist, jedem der 80 000 Hauptwörter von WordNet zwischen 500 und 1 000 Bilder manuell zuzuordnen. Aktuell⁶ sind 21 841 Wörter durch insgesamt 14 197 122 Bilder repräsentiert.

Neben den bisher vorgestellten großen und weit verbreiteten Datensätzen wurden auch kleinere bzw. weniger benutzte Datenbanken erstellt.

Bereits Mitte der 90-er Jahre wurden die sog. Columbia Object Image Libraries (COIL)-20⁷ [NNM96b] und COIL-100⁸ [NNM96a] erstellt. Beim COIL-20 wurden 20, beim COIL-100 100 verschiedene Objekte vor einem einfarbigen Hintergrund fotografiert.

1 <http://web.mit.edu/torralba/www/database.html>

2 <http://labelme.csail.mit.edu/>

3 <http://press.liacs.nl/mirflickr/>

4 <http://www.flickr.com>

5 <http://www.image-net.org>

6 Abruf vom 04.05.2012

7 <http://www1.cs.columbia.edu/CAVE/software/softlib/coil-20.php>

8 <http://www1.cs.columbia.edu/CAVE/software/softlib/coil-100.php>

Jedes Objekt wurde insgesamt 72 Mal abgelichtet mit je 5 Grad Drehung in der Horizontalen.

Zur Bewertung des Spatial-Envelope-Szenenbeschreibungsansatzes von [OT01] wurde ein eigener Datensatz¹ mit 8 verschiedenen Klassen erstellt. Dieser Datensatz wurde auch in [BZM08] verwendet.

In [SS04] wurde der Uncompressed Color Image Database (UCID)² erstellt, um das Problem des Corel Datensatzes zu umgehen und Forschungsergebnisse vergleichbar zu machen.

In [WAC⁺04] wurde die Xerox-Datenbank (auch bekannt unter dem Namen LAVA7) mit 1 776 Bildern verteilt auf 7 verschiedene Klassen zur Objekterkennung verwendet.

Der Datensatz der TU Darmstadt³ [LLS04] beinhaltet 326 annotierte Bilder in drei Kategorien: Seitenansichten von Motorrädern, Autos und Kühen.

Im Rahmen der in [OPFA06] vorgestellten Arbeit wurde an der TU Graz eine Bilddatenbank⁴ mit 3 verschiedenen Klassen und zusätzlichen Hintergrundbildern erstellt. Die Objekte variieren bzgl. ihrer Ausrichtung und Position innerhalb einer Klasse sehr stark. [Bil06] stellt den CBCL-Datensatz⁵ zur Kategorisierung von Straßenszenen bereit.

Microsoft Research Cambridge (MSRC)⁶ hat ebenfalls einen Bilddatensatz zum Download bereitgestellt.

Zur Klassifikation von Innenaufnahmen wurde in [QT09] ein Datensatz⁷ mit 67 verschiedenen Klassen erstellt.

Ein Vergleich der verschiedenen Datensätze ist in Tabelle 4.4 dargestellt. Da für den Vergleich von Objekterkennungsansätzen am häufigsten die Datensätze Caltech 101, Caltech 256 und PASCAL VOC 2007 eingesetzt werden, werden in dieser Arbeit diese 3 Datensätze verwendet. Zur Zeit besitzt nur der ImageNet-Datensatz mehrere 1 000 Klassen, deshalb wird in der vorliegenden Arbeit dieser Datensatz für die Evaluation der Skalierbarkeit und Erweiterbarkeit verwendet.

¹ <http://people.csail.mit.edu/torralba/code/spatialenvelope/>

² <http://vision.cs.aston.ac.uk/datasets/UCID/>

³ <http://www.vision.ee.ethz.ch/~bleibe/data/datasets.html>

⁴ http://www.emt.tugraz.at/~pinz/data/GRAZ_02/

⁵ <http://cbcl.mit.edu/software-datasets/streetscenes/>

⁶ <http://research.microsoft.com/en-us/downloads/b94de342-60dc-45d0-830b-9f6eff91b301/default.aspx>

⁷ <http://web.mit.edu/torralba/www/indoor.html>

DATENSATZ	JAHR	KLASSEN	BILDER
Caltech 101	2004	102	9 144
Caltech 256	2006	257	30 607
CBCL Straßenszenen	2004	9	3 547
COIL-20	1996	20	1 440
COIL-100	1996	100	7 200
Corel	–	800	80 000
Graz	2004	4	1 280
ImageNet	2012	21 841	14 197 122
LabelMe	2005	183	30 369
MIR Flickr	2008	30	25 000
MIR Flickr 1M	2010	30 (1 386)	1 000 000
MIT CSAIL	2004	107	2 873
MSRC	2005	23	591
[OT01]	2001	8	2 688
PASCAL VOC 2007	2007	20	5 011
[QT09]	2009	67	15 620
TU Darmstadt	2004	3	326
UCID	2004	–	1 338
Xerox7	2004	7	1 776

Tabelle 4.4: Vergleich von verschiedenen Datensätzen für die Evaluation von Objekterkennungsverfahren.

4.1.4 Bewertungsgrundlagen

Um verschiedene Ansätze der Objekterkennung vergleichen zu können, sind einheitliche Bewertungsgrundlagen nötig. Dabei können Bewertungsmaße zum Vergleich der Klassifikation und der Annotation unterschieden werden. In diesem Abschnitt werden die gängigsten Vergleichsmaße und Evaluationsfunktionen kurz vorgestellt.

4.1.4.1 Bewertungsmaße für die Klassifikation

In diesem Abschnitt werden verschiedene Maße zur Bewertung der Klassifikationsperformance erläutert.

Zuerst werden die wesentlichen Bestandteile aller Bewertungsmaße im Bildretrieval und der Objekterkennung nach [Hen08] und [DKN08] definiert. Relevante Dokumente, welche sich in der Ergebnismenge des betrachteten Verfahrens befinden, werden als richtig Positive (true positives, TP) bzw. „Hits“ bezeichnet. Diejenigen relevanten Dokumente,

welche es nicht in die Ergebnismenge geschafft haben, werden unter falsch Negativen (false negatives, FN) bzw. „Misses“ subsummiert. Richtig Negative (true negatives, TN) bzw. „Rejected“ sind nicht-relevante Dokumente, die auch nicht in der Ergebnismenge auftauchen. Diejenigen nicht-relevanten Dokumente, die dennoch in der Ergebnismenge auftauchen werden als falsch Positive (false positives, FP) bzw. „Noise“ bezeichnet. Die Anzahl der relevanten Dokumente ist also TP+FN, die Anzahl der Dokumente in der Ergebnismenge TP+FP.

Im Bildretrieval werden nach [Hen08] am häufigsten Precision und Recall zur Bewertung herangezogen. Precision spiegelt den Anteil der gefundenen relevanten Dokumente im gesamten Suchergebnis wieder. Precision ist somit ein Indiz dafür, inwieweit das System in der Lage ist nur relevante Ergebnisse zu liefern. Recall repräsentiert den Anteil der gefundenen relevanten Dokumente verglichen mit der Anzahl der relevanten Dokumente im gesamten Archiv. Bei beiden Werten sollte ein möglichst hoher Wert erreicht werden, damit möglichst alle relevanten Dokumente ins Suchergebnis kommen und dabei die Ergebnismenge möglichst ausschließlich aus relevanten Dokumenten besteht. Formal werden Precision (P) und Recall (R) nach [Hen08] und [DKN08] folgendermaßen definiert:

$$P = \frac{\text{Anzahl gefundener relevanter Dokumente}}{\text{Anzahl gefundener Dokumente insgesamt}} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (4.8)$$

$$R = \frac{\text{Anzahl gefundener relevanter Dokumente}}{\text{Anzahl relevanter Dokumente insgesamt}} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (4.9)$$

Da Precision und Recall sich gegenseitig beeinflussen, werden beide Werte häufig in einem Koordinatensystem als ein sog. PR-Graph dargestellt. Der Graph kann nach [Hen08] einerseits eingesetzt werden, um verschiedene Systeme basierend auf gleich vielen Top- N -Ergebnissen zu vergleichen (siehe Abbildung 4.11(a)). Andererseits kann auch zu jedem Recall-Wert zwischen 0 und 1 in 0,1-Schritten die Anzahl der nötigen Bilder und daraus die zugehörige Precision ermittelt werden. Anschließend kann aus den 11 Werten eine Kurve interpoliert und die Fläche – deren Wert im unerreichbaren Idealfall 1 beträgt – berechnet werden (siehe Abbildung 4.11(b)). Dieses Average Precision (AP) Verfahren kann je Anfrage oder auch für eine Menge von Anfragen eingesetzt werden. Bei Letzterem wird der Mittelwert der APs berechnet und wird als Mean Average Precision (MAP) bezeichnet. Das AP- bzw. MAP-Maß wird u. a. in [EVGW⁺07] und in

den ImageCLEF-Wettbewerben zur Bewertung und zum Vergleich von verschiedenen Objekterkennungsansätzen verwendet (vgl. Abschnitt 4.1.5).

Gleichung 4.10 zeigt die Definition von AP nach [DKN08], wobei q die spezifische Anfrage, N_R die Anzahl aller gefundenen Dokumente und R_n den Recall nach dem n -ten relevanten gefundenen Bild beschreibt. Die N_R Schritte werden für die Berechnung der AP üblicherweise auf die oben genannten 11 Abschnitte vereinfacht und die Kurve interpoliert. Gleichung 4.11 zeigt die Definition von MAP nach [DKN08], wobei Q die Menge der Anfragen darstellt.

$$AP(q) = \frac{1}{N_R} \sum_{n=1}^{N_R} P_q(R_n) \quad (4.10)$$

$$MAP = \frac{1}{|Q|} \sum_{q \in Q} AP(q) \quad (4.11)$$

Weitere Varianten sind auch die Fehlerrate nach [MMS⁺01] bzw. [DKN08] und das $1 - P(x)$ -Maß nach [DKN04]. Die Fehlerrate verallgemeinert die Relevanz von einzelnen Dokumenten auf Klassen, denen sie zugeordnet sind. Sie errechnet sich demnach wie in Gleichung 4.12 beschrieben. Beim $1 - P(x)$ -Maß steht $P(x)$ nach der Definition in [MMS⁺01] für den Precision-Wert nach dem x -ten zurückgelieferten Dokument.

$$ER = \frac{1}{|Q|} \sum_{q \in Q} \begin{cases} 0, & \text{falls das ähnlichste Bild relevant ist} \\ & \text{und aus der korrekten Klasse stammt} \\ 1, & \text{sonst.} \end{cases} \quad (4.12)$$

In [Rij76] wurde die Kombination von Precision (P) und Recall (R) als eine einzige Kennzahl eingeführt (siehe Gleichung 4.13). Hierbei ist es möglich Precision und Recall durch die entsprechende Wahl von α zu gewichten. Häufig wird die Gleichgewichtung der beiden Kennzahlen verwendet ($\alpha = 0,5$).

$$F - Measure = \frac{P \cdot R}{(1 - \alpha) \cdot R + \alpha \cdot P} \quad (4.13)$$

In [MMS⁺01] wurde neben den oben genannten Maßen auch der normalisierte durchschnittliche Rang der relevanten Bilder als Bewertungsgrundlage vorgeschlagen. In der

Definition in Gleichung 4.14 steht N_R für die Anzahl der relevanten Bilder, N für die Anzahl aller Bilder und R_i für den Rang des i -ten Bildes. Aus der Definition lässt sich ableiten, dass ein Verfahren umso besser ist, je näher der Wert für \widetilde{Rank} an 0 kommt.

$$\widetilde{Rank} = \frac{1}{NN_R} \left(\sum_{i=1}^{N_R} R_i - \frac{N_R(N_R - 1)}{2} \right) \quad (4.14)$$

Einer der am häufigsten verwendeten Bewertungsmethoden ist die Reciever Operating Characteristic (ROC) [Met78]. Hierzu werden auf der x -Achse die Anzahl der false positives (FP), auf der y -Achse die Anzahl der true positives (TP) aufgetragen. Zur Ermittlung der Bewertung werden bei verschiedenen Schwellwerten – oft in Abständen von 0,1-Schritten – Messungen bzgl. TP und FP durchgeführt (siehe Abbildung 4.11(c)). Eine objektive Messgröße für den Vergleich von unterschiedlichen Verfahren stellt dabei die Fläche unterhalb der ROC Kurve, die Area Under the ROC Curve dar.

Häufig wird auch der Zusammenhang zwischen der Anzahl von Trainingsbildern und der erreichten durchschnittlichen Erkennungsrate je Klasse als Vergleichskriterium herangezogen. Abbildung 4.11(d) stellt ein Beispiel dar.

Die sog. Confusion Matrix wird zur Visualisierung der Genauigkeit der Klassifikation eines Ansatzes verwendet. Hierbei werden alle Klassen sowohl auf der x - als auch auf der y -Achse aufgetragen. Die Zellen beschreiben prozentual wie viele Elemente einer Klasse Ω_i der Klasse Ω_j zugeordnet wurden. Ein Beispiel für eine Confusion Matrix ist in Abbildung 4.11(e) dargestellt. Im Idealfall steht auf der Querlinie überall 100% und sonst 0%. Der Durchschnitt der Werte auf der Querlinie kann auch als die MAP aufgefasst werden (vgl. Gleichung 4.11).

4.1.4.2 Bewertungsmaße für die Annotation

Neben den Maßen zur Bewertung der Klassifikation wurden auch spezielle Evaluationsfunktionen für den objektiven Vergleich verschiedener Annotationssysteme entwickelt. In diesem Abschnitt werden Bewertungsmaße für die textuelle Annotation von Bildern (bzw. allgemein für multimediale Objekte) vorgestellt.

In [NL09] wurde ein neues Bewertungsmaß für die Evaluation von Multilabel-Klassifikationsansätzen vorgeschlagen, welches auch im ImageCLEF Fotoannotations-

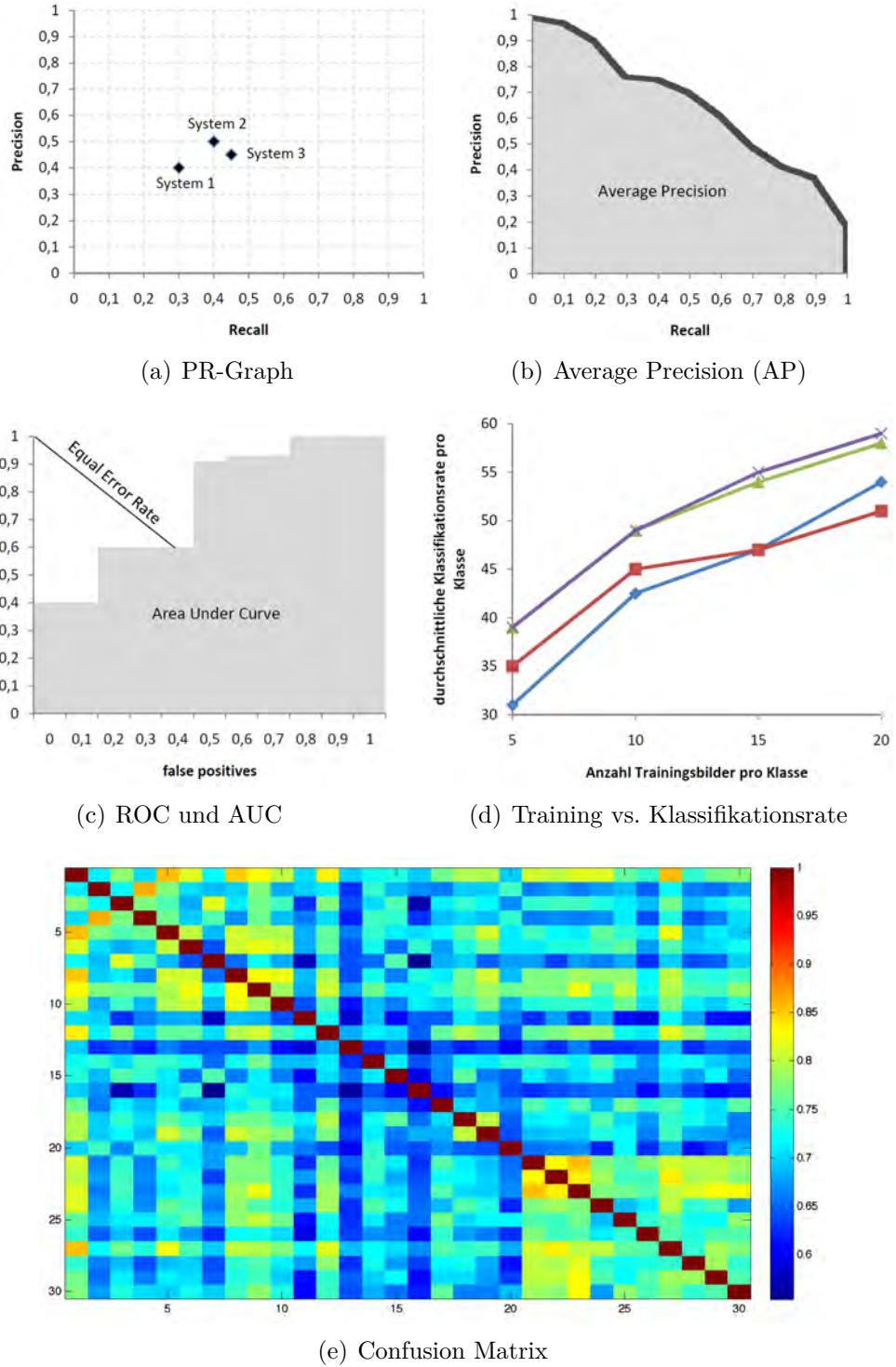


Bild 4.11: Darstellung von gängigen Bewertungsmaßen im Bildretrieval und der Objektkennung.

wettbewerb 2009¹ eingesetzt wurde. Bei diesem Bewertungsmaß werden die Beziehungen zwischen den Wörtern (bzw. Konzepten) basierend auf einer Ontologie bzw. einer Worthierarchie mit berücksichtigt. Dabei kann nur eine Ontologie und immer nur ein hierarchischer Beziehungstyp berücksichtigt werden.

Im Folgenden sei P die Menge der ermittelten, G die Menge der erwarteten Konzepte. Die Elemente, das heißt die einzelnen Konzepte der jeweiligen Mengen, werden mit C_p bzw. C_g gekennzeichnet. Zuerst werden die Mengen der false positives $P' = P \setminus (P \cap G)$ und der false negatives $G' = G \setminus (P \cap G)$ ermittelt. Anschließend werden die Kosten $cost(C_p, C_g)$ zwischen jedem Paar $\{(C_p, C_g) \mid C_p \in P', C_g \in G\}$ und $\{(C_g, C_p) \mid C_g \in G', C_p \in P\}$ berechnet. Die Kosten ergeben sich dabei aus dem Abstand der zwei Wörter C_p und C_g in der verwendeten Ontologie, wobei sich die Kosten für die Kante i nach der Gleichung 4.15 berechnen. N kennzeichnet dabei die Anzahl der Kanten vom tiefsten Knoten bis zur Wurzel.

$$c_i = \frac{2^{(i-1)}}{2 \cdot \sum_{i=1}^N 2^{(i-1)}} \quad (4.15)$$

Die Übereinstimmungskosten können nach

$$match(P, G) = \sum_{C_p \in P'} \left(\min_{C_g \in G} cost(C_p, C_g) \cdot a(C_g^*) \right) + \sum_{C_g \in G'} \left(\min_{C_p \in P} cost(C_p, C_g) \cdot a(C_g) \right) \quad (4.16)$$

berechnet werden, wobei $C_g^* = \arg \min_{C_g \in G} (cost(C_p, C_g))$ und $a(C_g) \in [0,1]$ der empirisch durch annotierende Personen festgelegte Annotations-Zustimmungsfaktor für das Konzept $C_g \in G$ ist. Sofern ein Konzept $C_p \in P$ in der Ontologie nicht vorkommt, wird es mit $match(P, G) = 1$ bewertet. Für ein multimediales Objekt x ergibt sich die Annotationspunktzahl nach

$$score(x) = \left(1 - \frac{match(P, G)}{|P \cup G|} \right)^\alpha, \quad (4.17)$$

wobei durch den Parameter α die Strenge (bzw. nach [SBLB04] die „Vergebungsrate“) der Bepunktung feinjustiert werden kann. Je näher die Annotation P der erwarteten Konzeptmenge G ist, desto näher kommt der Wert $score(x)$ an 1. Für den gesamten

¹ <http://www.imageclef.org/2009/PhotoAnnotation>

Datensatz D ergibt sich damit die Genauigkeit der Annotationen nach Gleichung 4.18.

$$accuracy_D = \frac{1}{|D|} \sum_{x \in D} score(x) \quad (4.18)$$

[SBLB04] verwendet ein ähnliches Maß für die Bewertung der Annotationen, welches in Gleichung 4.19 dargestellt ist. Bei diesem Bewertungsmaß wird keine Ontologie vorausgesetzt, es werden lediglich die durch das gegebene Verfahren annotierten Konzepte mit den erwarteten Konzepten verglichen.

$$score(x) = \left(1 - \frac{|P \cap G|}{|P \cup G|}\right)^\alpha \quad (4.19)$$

Weitere Maße wurden in [SL01] und [CBGZ06] definiert, diese werden jedoch wegen ihrer niedrigen Akzeptanz (bislang keine Verwendung in Wettbewerben oder wissenschaftlichen Veröffentlichungen) hier nicht vorgestellt.

4.1.4.3 Ähnlichkeitsmetriken für ontologiebasierte Bewertungsmaße

An der Stelle der *cost*-Funktion in Abschnitt 4.1.4.2 können auch andere Ähnlichkeits- bzw. Distanzmaße für hierarchische Ontologien eingesetzt werden. Die Ähnlichkeitsmaße $sim(C_a, C_b)$ von Konzepten C_a und C_b können durch Umrechnung der Distanzmaße $dist(C_a, C_b)$ mittels $1 - dist(C_a, C_b)$ (nach [SZF07]) oder $\frac{1}{1+dist(C_a, C_b)}$ (nach [Lin98]) konvertiert werden.

Die Ähnlichkeitsmaße für Konzepte in hierarchischen Ontologien können auf kantenbasierte, knotenbasierte und hybride Ansätze aufgeteilt werden. Im nachfolgenden werden die verschiedenen Ähnlichkeitsmaße kurz beschrieben, wobei nur diejenigen im Detail erläutert werden, welche auch in der Objekterkennung bislang eingesetzt wurden. Die vollständige Darstellung der Ähnlichkeitsmaße ist in Anhang C aufgeführt.

Kantenbasierte Ähnlichkeitsmaße

Kantenbasierte Ähnlichkeitsmaße bestimmen die Distanz zwischen zwei Konzepten durch die Anzahl der Kanten $lenE(C_a, C_b)$ die in der hierarchischen Ontologie nötig sind, um auf dem kürzesten Pfad vom Konzept C_a zum Konzept C_b zu gelangen. Beispiele sind die

in Abschnitt 4.1.4.2 aus [NL09] vorgestellte Kostenfunktion sowie Ansätze aus [Res95] und [LC98], die in Anhang C.1 näher vorgestellt werden.

Knotenbasierte Ähnlichkeitsmaße

Bei knotenbasierten Ähnlichkeitsmaßen wird die Distanz $lenN(C_a, C_b)$ zwischen Konzepten C_a und C_b durch die Anzahl der Knoten auf dem kürzesten Pfad in der hierarchischen Ontologie bestimmt. Einfache Ansätze wurden in [WP94] und [Res95] beschrieben. [Lin98], [JC97] und [SDRL06] integrierten zusätzlich den Informationsgehalt von Konzepten in ihre Ähnlichkeitsmaße. Detaillierte Beschreibungen dieser Ansätze sind in Anhang C.2 zu finden.

Für den Vergleich von verschiedenen Objekterkennungsansätzen wurde beim ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2011 Wettbewerb ebenfalls ein knotenbasiertes Ähnlichkeitsmaß eingesetzt. Das Ziel bei diesem Maß war die Berechnung des hierarchischen Fehlers für ein gegebenes Bild. Dabei wurde die Distanz zwischen zwei Konzepten C_a und C_b wie folgt definiert:

$$dist(C_a, C_b) = \frac{h(LCA(C_a, C_b))}{\max(h(C_a), h(C_b))}, \quad (4.20)$$

wobei $LCA(C_a, C_b)$ der nächste gemeinsame Elternknoten von C_a und C_b in der verwendeten Kategorienhierarchie ist und $h(C_k)$ die Anzahl der Knoten bis zur Wurzel (die Höhe des Knotens C_k in der Hierarchie, also $h(C_k) = lenN(C_k, C_r)$ mit C_r als Wurzel) berechnet.

Der Fehler für ein Bild I kann anschließend durch folgende Berechnung ermittelt werden:

$$error(I) = \frac{1}{n} \sum_{k=1}^n \arg \min_j dist(C_{g_k}, C_{w_j}). \quad (4.21)$$

Im ILSVRC 2011 Wettbewerb wurden dabei die ersten 5 vom Klassifikator ermittelten Konzepte $C_{w_j}, j = 1, \dots, 5$ herangezogen und die kleinsten hierarchischen Fehlerwerte je Grundwahrheitsterm $C_{g_k}, k = 1, \dots, n$ gemittelt. Da bei ImageNet die vorgegebene Annotation immer nur ein einziges Konzept ist (also $n = 1$), kann die Formel zur

Berechnung des hierarchischen Fehlers für ein gegebenes Bild I wie in Gleichung 4.22 vereinfacht werden.

$$error(I) = \arg \min_j dist(C_g, C_{w_j}) \quad (4.22)$$

Zur Berechnung des hierarchischen Fehlers für alle Testbilder $I \in S$ einer Kategorie wird der Mittelwert über alle hierarchischen Fehler $error(I)$ der Testbilder berechnet.

$$mean_error(S) = \frac{1}{|S|} \sum_{I \in S} error(I), \quad (4.23)$$

wobei $|S|$ für die Anzahl der Testbilder steht.

Hybride Ähnlichkeitsmaße

In [SZF07] wurde ein Ähnlichkeitsmaß definiert, welches für jedes Konzept a priori Kosten berücksichtigt, die die erwarteten Kosten eines durchschnittlichen Benutzers erfassen. Die Berechnung dieser a priori Kosten basiert auf der Wahrscheinlichkeit, ob ein Konzept einem anderem übergeordnet ist. Die detaillierte Beschreibung des Ähnlichkeitsmaßes ist in Anhang C.3 zu finden.

4.1.4.4 Zusammenfassung

In diesem Abschnitt wurden Bewertungsmaße zum Vergleich der Klassifikationsperformance und der Annotationsgenauigkeit von verschiedenen Ansätzen vorgestellt. In den Wettbewerben und in den Veröffentlichungen wurde bislang am häufigsten AUC und MAP als Bewertungsmaß eingesetzt. Zur Evaluation von Annotationen unter Berücksichtigung der Beziehungen zwischen Wörtern basierend auf einer Ontologie werden zur Zeit die Bewertungsmaße nach den Gleichungen 4.18 und 4.23 eingesetzt. Bei den meisten ontologiebasierten Bewertungsmaßen fehlt zur Zeit noch die Verallgemeinerung auf die Verwendung von mehreren verschiedenen Ontologien bzw. mehreren Beziehungstypen zwischen Wörtern (vgl. Anhang B).

4.1.5 Wettbewerbe

Um die Forschung u. a. auf dem Gebiet der Objekterkennung voranzutreiben, sind mehrere Wettbewerbe (Challenges) entstanden, welche jedes Jahr basierend auf einem festgelegten Datensatz und unter Verwendung der in Abschnitt 4.1.4 vorgestellten Bewertungsmaße den besten Ansatz ermitteln. Die größten drei Wettbewerbe auf dem Gebiet der Objekterkennung sind PASCAL VOC, ImageCLEF und ILSVRC, welche in diesem Abschnitt kurz vorgestellt werden.

Das PASCAL Netzwerk¹ veranstaltet jedes Jahr mehrere verschiedene Wettbewerbe, unter anderem auch den Wettbewerb PASCAL VOC zur Objekterkennung, Segmentierung und zur Erkennung von menschlichen Gliedmaßen in Bildern. Für jedes Teilgebiet wird jeweils ein Bilddatensatz zur Verfügung gestellt. Zur Objekterkennung wird der Datensatz auf eine Trainings-, eine Validierungs- und eine Testmenge aufgebrochen. Seit dem ersten PASCAL VOC in 2005 wurde der Datensatz stetig erweitert. Die Entwicklung ist in Tabelle 4.5 veranschaulicht. Als Bewertungsmaß diente bei den ersten Wettbewerben noch AUC, mittlerweile wird jedoch nur noch MAP herangezogen. Die Ergebnisse der jährlich stattfindenden Wettbewerbe werden auf Workshops bei großen Computer Vision Konferenzen (wie z. B. ICCV oder ECCV) bekanntgegeben.

JAHR	KLASSEN	BILDER	BERICHT
2005	4	2 232	[EZW ⁺⁰⁵]
2006	10	2 618	[EZWVG06]
2007	20	5 011	[EVGW ⁺⁰⁷]
2008	20	4 340	[EVGW ⁺⁰⁸]
2009	20	7 054	[EVGW ⁺⁰⁹]
2010	20	10 103	[EVGW ⁺¹⁰]
2011	20	11 540	[EVGW ⁺¹¹]

Tabelle 4.5: Entwicklung der PASCAL VOC Datensätze für die Objekterkennungsaufgabe.

Der ImageCLEF² Wettbewerb wurde vor allem für das Gebiet des Bildretrievals gegründet und findet seit 2003 statt. Gegenstand der verschiedenen und von Jahr zu Jahr wechselnden Aufgaben sind Fotografien und medizinische Bilder mit oder ohne Annotationen. Im Jahr 2005 wurde das erste Mal eine Aufgabe zur automatischen Annotation von medizinischen Bildern gestellt. 2006 wurde dieser Wettbewerb erneut organisiert und um

1 <http://www.pascal-network.org>

2 <http://www.imageclef.org>

die automatische Annotation von Fotografien erweitert. Die Annotation beschränkte sich auf insgesamt 21 Objektklassen. Da sowohl das Interesse an diesem Annotationswettbewerb sehr gering war, als auch die insgesamt 3 Teilnehmer nur recht dürftige Ergebnisse erzielen konnten, wurde dieser Annotationswettbewerb in den nachfolgenden Jahren nicht erneut organisiert. 2007 wurde das erste Mal ein Wettbewerb zur Ermittlung des besten Objekterkennungsansatzes veranstaltet. Hierbei mussten Bilder allein basierend auf visuellen Objekterkennungstechniken 10 verschiedenen Klassen zugeordnet werden. Seit 2008 findet jedes Jahr ein Wettbewerb zur Erkennung von Konzepten in Bildern statt. Hierbei sollen z. B. Innen- von Außenaufnahmen oder Szenen wie Küstengebiete von Gebäuden und Bergen unterschieden werden. Als Grundlage wird eine Teilmenge des MIR Flickr 25 000 bzw. des MIR Flickr 1M Datensatzes verwendet. Für die Bewertung der Verfahren wird neben AUC und MAP der hierarchische Abstand nach Gleichung C.11 mittels WordNet berechnet.

Seit 2010 findet unter dem Namen ILSVRC auch ein groß angelegter Wettbewerb auf einer Teilmenge des ImageNet-Datensatzes statt. Ziel des ILSVRC-Wettbewerbs ist es Objekterkennungsverfahren in realitätsnäheren Dimensionen zu evaluieren und somit die Forschung auf dem Gebiet der Skalierbarkeit der Ansätze voranzutreiben. Dementsprechend wurden Bilder zu 1 000 unterschiedlichen Konzepten bereitgestellt, wobei alle Konzepte Blätter in der WordNet-Hierarchie sind. Zur Zeit werden bei der Bewertung der Objekterkennungsverfahren die besten 5 Konzepte je Bild berücksichtigt. Als Bewertungsmaß wird der hierarchische Fehler nach Gleichung 4.23 mittels WordNet berechnet. Die Ergebnisse der Wettbewerbe werden auf den jährlich stattfindenden PASCAL Workshops bekanntgegeben [BDL10, BDL11].

Tabelle 4.6 fasst die verschiedenen Wettbewerbe auf dem Gebiet der Objekterkennung zusammen und gibt einen Überblick über die verwendeten Bewertungsmaße. Für den Vergleich der eingesetzten Anzahl der Bilder sowie der verwendeten Anzahl von Klassen wurden die Wettbewerbe aus dem Jahr 2011 herangezogen.

NAME	BEWERTUNGSMASS	KLASSEN	BILDER
PASCAL VOC	(AUC), MAP	20	11 540
ImageCLEF	AUC, MAP, hierachischer Abstand	99	18 000
ILSVRC	hierachischer Fehler	1 000	1 200 000

Tabelle 4.6: Vergleich der Objekterkennungswettbewerbe PASCAL VOC, ImageCLEF und ILSVRC.

4.1.6 Aktuelle Objekterkennungsverfahren

In diesem Abschnitt werden die besten Objekterkennungsverfahren als Beispiel für den Einsatz des im Abschnitt 4.1.2 beschriebenen BoW-Konzepts vorgestellt. Da in den PASCAL-Wettbewerben und bei Messungen auf den Caltech-Datensätzen diskriminative Ansätze am besten abgeschnitten haben, werden in Abschnitt 4.1.6.1 mehrere erfolgreiche diskriminative Ansätze kurz vorgestellt. Nachfolgend werden diskriminative, jedoch auch erweiterbare Ansätze in Abschnitt 4.1.6.2 beschrieben. Beispiele für generative Verfahren werden in Abschnitt 4.1.6.3 kurz erläutert. Abschließend werden die Ergebnisse in Abschnitt 4.1.6.4 zusammengefasst.

4.1.6.1 Diskriminative Ansätze

[ZMLS07] erwies sich als bester Ansatz bei der Klassifikationsaufgabe von PASCAL VOC 2005 [EZW⁺05]. Zur Ermittlung der interessanten Punkte wurde ein Ecken-ähnlicher Harris-Laplace und ein Blob-ähnlicher Laplacian Detektor verwendet. Die ermittelten Punkte wurden anschließend durch SIFT, Spin und Rotation-Invariant Feature Transform (RIFT) Deskriptoren beschrieben. Aus diesen Merkmalen wurde durch k -means Clustering das Vokabular der visuellen Wörter ermittelt. Für den Vergleich der Histogramme von visuellen Wörtern wurde die EMD und die χ^2 Distanz verwendet. Diese Distanzmaße wurden auch als Kernfunktionen in SVMs für die Klassifikation von Bildern verwendet. Der Ansatz wurde anschließend auf mehreren Bilddatensätzen getestet, wobei das Verfahren sowohl bei reinen Texturaufnahmen als auch bei Bildern von Objekten mit variierenden Hintergründen sehr gut abschnitt.

Das Verfahren aus [ZMLS07] wurde im PASCAL-VOC-2006-Wettbewerb um den Spatial-Pyramid-Ansatz aus [LSP06] erweitert. Das neue QMUL¹ genannte Verfahren [EZWVG06] verwendet einen zweischichtigen SVM-Klassifikator. Hierbei werden zuerst für jede Ebene im Spatial Pyramid SVM-Klassifikatoren trainiert (erste Schicht). Basierend auf deren Ausgaben wird ein weiterer SVM-Klassifikator erlernt (zweite Schicht). Die Erweiterung durch den Spatial-Pyramid-Ansatz erwies sich als erfolgreich, da dieser neue Ansatz das beste Verfahren im PASCAL-VOC-2006-Wettbewerb war.

Ähnlich erfolgreich schneidet im PASCAL-VOC-2006-Wettbewerb der in [PDCB06] näher vorgestellte Xerox Research Center Europe (XRCE)-Ansatz ab. Bei diesem Verfahren

¹ benannt nach Queen Mary University of London

wird ein allgemeines Vokabular für alle Klassen sowie Klassen-Vokabulare, welche das allgemeine Vokabular basierend auf klassenspezifischen Daten adaptieren, verwendet. Ein Bild wird demnach durch eine Menge von Histogrammen beschrieben (je ein Histogramm pro Klasse). Zur Erstellung der Vokabulare werden alle Bilder auf die gleiche Größe skaliert und anschließend mittels festgesetzter Gitternetze Punkte ermittelt (dense Sampling). Diese Punkte werden durch SIFT-Deskriptoren beschrieben und anschließend mittels der Hauptkomponentenanalyse (Principal Component Analysis (PCA), [DHS00]) von 128 auf 50 Dimensionen reduziert. Die Vokabulare werden durch ein GMM und durch die Maximum Likelihood Estimation (MLE) iterativ berechnet. Die Anzahl der Wörter der Vokabulare kann durch die iterative Erweiterung des Vokabulars durch weitere Gauß-Kurven angepasst werden. Anschließend wird je Klasse das klassenspezifische und das allgemeine Vokabular zusammengefügt, wie in Abbildung 4.12 veranschaulicht. Durch diese Aufteilung bzw. Vermischung werden relevante Informationen (klassenspezifisches Vokabular) von irrelevanten Informationen (allgemeines Vokabular) getrennt.

Bei einem neuen zu klassifizierenden Bild wird basierend auf jedem Vokabular je Klasse ein bipartites Histogramm berechnet, das zusammen mit den Histogrammen der anderen Klassen sequentiell den klassenspezifischen linearen SVM-Klassifikatoren zugeführt wird. Prinzipiell heißt das, dass, sofern ein Bild durch das allgemeine Vokabular besser beschrieben wird, es mit hoher Wahrscheinlichkeit nicht zu der entsprechenden Klasse gehört. Sofern das Bild jedoch vom klassenspezifischen Vokabular besser beschrieben wird, gehört das Bild mit hoher Wahrscheinlichkeit zu der entsprechenden Klasse (Abbildung 4.13).

Das Verfahren aus [PDCB06] wurde in [PD07] und [Per08] erweitert. Die durch ein Gitternetz ermittelten Punkte wurden neben SIFT-Deskriptoren auch durch lokale RGB-Statistiken beschrieben. Bei den RGB-Statistiken wird der zu beschreibende Bildpunkt (Bildausschnitt in Quadratform) auf $4 \times 4 = 16$ Regionen aufgeteilt und anschließend jeweils der Mittelwert und die Standardabweichung pro Farbkanal berechnet. Beide Merkmale wurden durch PCA jeweils von 128 bzw. 96 auf jeweils 50 Dimensionen reduziert. Zur Berechnung des Vokabulars wurden Fisher-Kernfunktionen nach [JH98] für das gegebene Szenario angepasst. Die entsprechenden Berechnungen sind in [PD07] zu finden. Durch die Verwendung von Fisher-Kernfunktionen wurde die Größe des Vokabulars um das 100-fache geschrumpft und die Berechnungszeit erheblich verkürzt. Trotz dieser Einsparungen hat sich die Klassifikationsperformance bzgl. [PDCB06] nicht verschlechtert. Interessant bei dem Ansatz von [Per08] ist, dass er die Erstellung einer Hierarchie ermöglicht. Das Verfahren schafft beim ILSVRC 2011 Wettbewerb unter den zwei Besten ab.

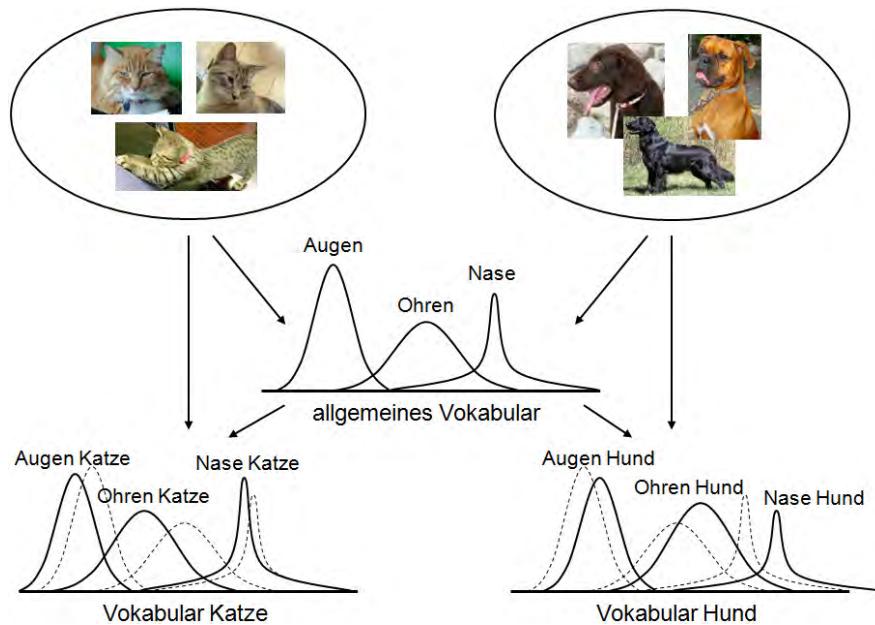


Bild 4.12: Allgemeine und klassenspezifische Vokabulare nach [PDCB06].

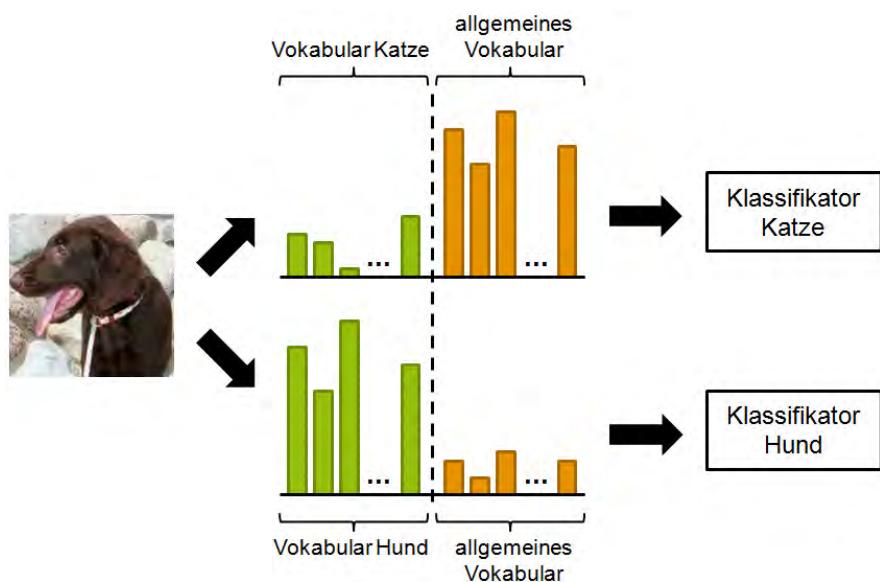


Bild 4.13: Bipartite Histogramme und klassenspezifische Klassifikatoren nach [PDCB06].

Beim PASCAL VOC 2007 Wettbewerb schnitt das Verfahren von [MSHW07] am besten ab. Dieser Ansatz ist im wesentlichen eine Erweiterung des QMUL-Verfahrens. Zur Ermittlung der interessanten Punkte wurden das Harris-Laplace- (Kantenerkennung), das Laplacian-Verfahren (Bloberkennung) sowie mehrere feste Gitter (dense Sampling) eingesetzt. Die ermittelten Punkte wurden anschließend durch SIFT-, hueSIFT- und Pairs of Adjacent Segments (PAS) Deskriptoren beschrieben. Um das visuelle Vokabular zu erhalten, wurden mittels eines k -means Clustering Algorithmus die ermittelten Merkmale auf 4 000 Cluster aufgeteilt. Der Spatial-Pyramid-Ansatz aus [LSP06] wurde ebenfalls integriert, indem die Merkmale drei verschiedenen räumlichen Ebenen zugeordnet wurden: das komplette Bild (1x1), Aufteilung in 4 gleich große Quadranten (2x2) und horizontale Aufteilung in drei Regionen (3x1). Die Kombination dieser Möglichkeiten (Detektoren verknüpft mit Deskriptoren und räumlicher Aufteilung, z. B. Harris-Laplace mit hueSIFT und 4 Quadranten) wurde in [MSHW07] zum ersten Mal als „Kanal“ bezeichnet (siehe Abschnitt 4.1.2.4). Zur Klassifikation wurde ein SVM mit einer multikanälig erweiterten Gauß-Kernfunktion eingesetzt. Als Distanzmaß wurde χ^2 verwendet. Zum Erlernen der Parameter des SVM wurde ein genetischer Algorithmus eingesetzt. Bei der Evaluation des Ansatzes wurden verschiedene Kanäle nacheinander hinzugefügt und die Veränderung der AP beobachtet. Wesentliche Erkenntnisse hieraus waren, dass die Kombination von sparse und dense Sampling, die zusätzliche Aufteilung auf drei horizontale Regionen und die Hinzunahme der Farbinformationen (hueSIFT) die AP im Durchschnitt verbessern.

Der Ansatz aus [MSHW07] wurde für den PASCAL VOC 2008 Wettbewerb weiterentwickelt. Das in [GMS08] vorgestellte Verfahren verwendet ebenfalls unterschiedliche Kanäle (Kombinationen aus Detektoren, Deskriptoren und Gitternetzen). Neben dense Sampling, Harris-Laplace und Laplacian wurden auch Hessian und Harris-Harris Detektoren eingesetzt. Die ermittelten Punkte wurden durch SIFT- und OpponentSIFT-Deskriptoren beschrieben. Die Gitternetze sind ähnlich geblieben wie in [MSHW07] (1x1, 2x2, 3x1). Für den Vergleich von Bildern wurde auch hier das χ^2 Distanzmaß verwendet. Die Klassifikation erfolgte durch zwei verschiedene SVM-Lösungen. Bei einer Methoden wurde das Verfahren aus [ZMLS07] verwendet. Bei der anderen Methode wurden die Parameter der allgemeinen Radial Basis Function (RBF)-Kernfunktion (ein Kern pro Klasse) durch den shotgun hill climbing Algorithmus (siehe Abschnitt 5.1.4.1) berechnet. Das gesamte Vorgehen von [GMS08] schnitt beim PASCAL VOC 2008 Wettbewerb gemeinsam mit [TSU⁺08] als bestes Verfahren ab.

Das Verfahren von [TSU⁺08] basiert ebenfalls auf den unterschiedlichen Kombinationen von Detektoren, Deskriptoren und Gitternetzen. Abbildung 4.14 veranschaulicht die verschiedenen Kanäle in [TSU⁺08]. Als Detektoren wurden Harris-Laplace und dense Sampling eingesetzt. Die ermittelten Punkte wurden durch SIFT-, OpponentSIFT-, WSIFT-, rgSIFT- und transformed color SIFT Deskriptoren beschrieben. Die Aufteilung des Bildes erfolgte ähnlich wie in [MSHW07] (1x1, 2x2, 3x1). Anschließend wurde mittels k -means ein visuelles Vokabular mit 4 000 Clustern berechnet. Für die Zuordnung von visuellen Wörtern zu Clustern wurden die einzelnen Cluster nach dem Verfahren von [GGVS08] durch Gauß-Kerne beschrieben. Zur Klassifikation wurde statt einem SVM das Spectral Regression Kernel Discriminant Analysis (SRKDA) Verfahren aus [CHH07] verwendet, welches die spektrale Graphanalyse mit Regression kombiniert, um das Problem der Kernel Discriminant Analysis (KDA) effizient zu lösen. Das Verfahren schnitt auch beim ILSVRC 2011 Wettbewerb unter den zwei Besten ab.

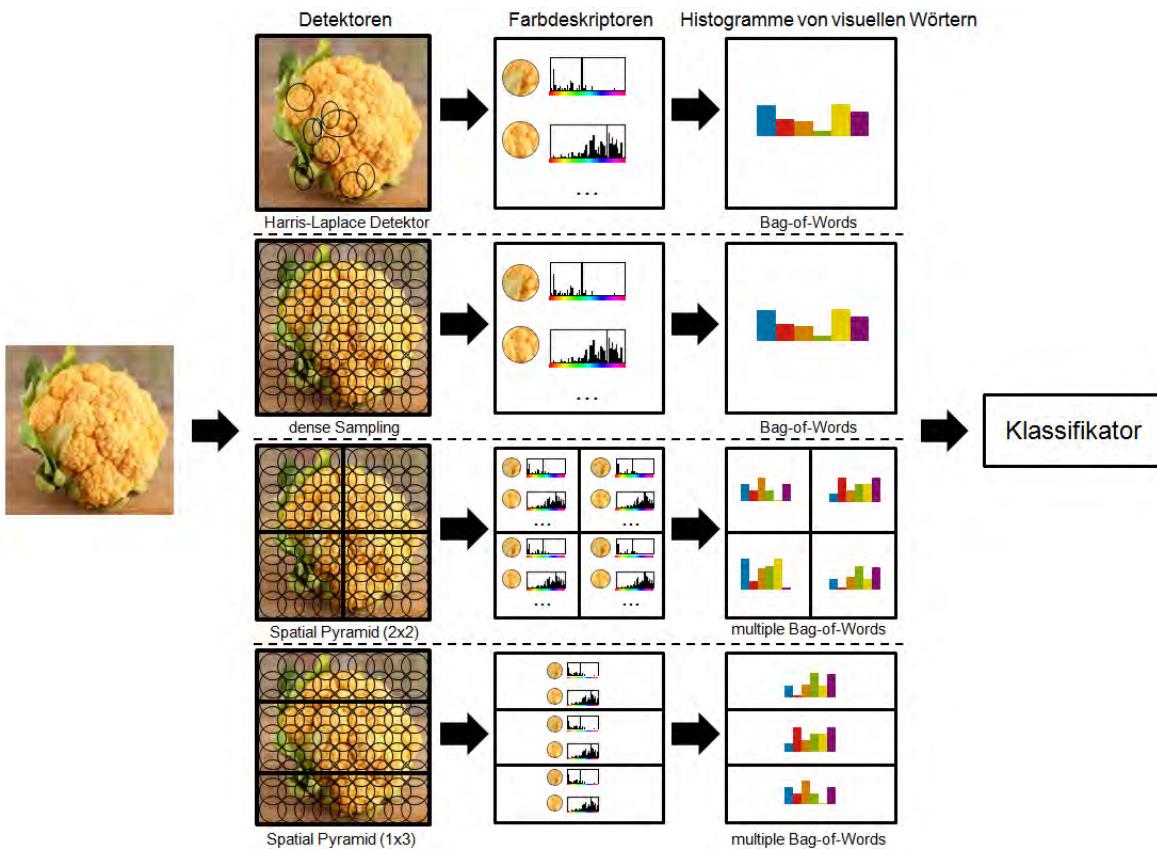


Bild 4.14: Objekterkennungsverfahren mit mehreren Kanälen und Farb-SIFT-Deskriptoren nach [TSU⁺08].

Basierend auf den Ergebnissen von [BLJ04] wurde in [KS07] die sog. Support Kernel Machine (SKM) zur Objekterkennung eingesetzt. SKMs ermöglichen die Kombination

mehrerer verschiedener Kernfunktionen und entsprechen dem Multiple Kernel Learning (MKL). Im Ansatz von [KS07] wurde die Spatial-Pyramid-Kernfunktion aus [LSP06] und die Pyramid-Match-Kernfunktion aus [GD05] verwendet. Mittels festgelegter Gitternetze wurden Punkte aus Bildern extrahiert und anschließend durch Farbhistogramme, SIFT- und PCA-SIFT-Deskriptoren beschrieben. Für die farbbasierten Kerne wurde der Spatial-Pyramid-Ansatz verwendet. Das Verfahren wurde u. a. auf den Caltech 101 und Caltech 256 Datensätzen evaluiert. Untersucht wurde, inwieweit die durchschnittliche Erkennungsrate pro Objektklasse von der Kombination der Kerne und der Anzahl der Trainingsbilder pro Klasse abhängt, bzw. wie die Verteilung der Gewichte der verwendeten Kerne variiert. Die wesentliche Erkenntnis war, dass SKMs eine skalierbare Lösung für die Kombination einer großen Anzahl von Kernen in heterogenen Merkmalsräumen ermöglichen, wo die a priori Gewichtung allein durch Intuition nicht mehr möglich ist.

Einen ähnlichen Ansatz wie in [KS07] zur Kombination von verschiedenen Kernfunktionen mittels MKL verfolgt auch [VR07]. Für den Caltech 101 Datensatz wurden mittels festen Gitternetzen Punkte ermittelt und anschließend durch SIFT-, HSV-SIFT-, GB- (nach [BBM05]) und PHOG-Deskriptoren (nach [BZM07]) beschrieben. Anschließend wurde ein visuelles Vokabular erstellt. Zur Ermittlung der Ähnlichkeit zwischen zwei Bildern wurde die Spatial-Pyramid-Kernfunktion aus [LSP06] verwendet. Zuletzt wurden SVMs mittels MKL mit optimaler Gewichtung der einzelnen Kerne berechnet. Bei der Evaluation des Verfahrens auf dem Caltech 101 Datensatz wurde eine bessere Klassifikationsrate erreicht als bei dem derzeitigen besten Ansatz. Ein wesentlicher Vorteil des Verfahrens von [VR07] ist, dass es mit verschiedenen Merkmalen zurechtkommt. Diese Eigenschaft wurde durch Messungen auf verschiedenen anderen Bilddatensätzen demonstriert (UIUC Textur- und Oxford Blumendatensatz).

[BR07] integriert in ihrem Verfahren viele der erfolgreichen Methoden. Zur Ermittlung der interessanten Punkte wurde ein festes Gitternetz und der Canny Kantendetektor in Verbindung mit einer Sobel-Maske verwendet. Diese Punkte wurden anschließend einerseits durch SIFT- und HSV-SIFT-Deskriptoren beschrieben, welche in [BR07] in Anlehnung an [LSP06] Pyramid Histogram of Words (PHOW) benannt wurden. Andererseits wurden die Kantenbilder durch Pyramid Histogram of Oriented Gradients (PHOG) Deskriptoren beschrieben, welche den Spatial-Pyramid-Ansatz aus [LSP06] und den Histogram of Gradients (HoG) Deskriptor aus [DT05] verbinden. Außerdem wurden noch weitere Deskriptoren, wie z. B. GIST und der Geometric Blur Deskriptor (GB) verwendet. Zusätzlich wurde basierend auf den Merkmalen auch der Region of Interest (ROI)

ermittelt. Die Kombination von verschiedenen Kernen basierte auf den Erkenntnissen von [KS07] und [VR07]. Als Distanzmaß wurde χ^2 verwendet. Die Klassifikation erfolgte mittels SVMs. Diese Methode hat auf dem Caltech 101 und Caltech 256 Datensatz das beste Ergebnis von allen Ansätzen erzielt. Erkenntnisse dieses Verfahrens waren, dass die am meisten diskriminativen Merkmale PHOW waren, dicht gefolgt von PHOG. Die Ermittlung der ROI verbessert die Klassifikationsrate um 2 bis 5% und das klassenspezifische Erlernen der Gewichte erhöht dies nochmal um 4%.

4.1.6.2 Erweiterbare Ansätze

Die bislang vorgestellten Verfahren sind nicht erweiterbar, da sie entweder auf einem globalen Vokabular aufbauen oder SVMs als Klassifikatoren einsetzen. Eine detaillierte Begründung hierzu wird in Abschnitt 5.1 gegeben.

In [BSI08] wurde ein erweiterbares Verfahren vorgestellt, welches auf den Caltech 101, Caltech 256 und Graz Datensätzen ähnlich gut oder besser als die oben genannten Verfahren abschnitt. Die Bilder werden zuerst mittels dense Sampling abgetastet und durch SIFT-Deskriptoren beschrieben. Des Weiteren werden auch einfache Farb- und Formdeskriptoren verwendet. Aus den SIFT-Deskriptoren wurde kein visuelles Vokabular erstellt, was wesentlich zur Erweiterbarkeit beiträgt. In der Veröffentlichung wurde auch darauf eingegangen, dass die Quantisierung der SIFT-Deskriptoren in visuelle Wörter deren Diskriminativität erheblich verringert. Weiterhin wird statt SVMs ein NN-Klassifikator mit der euklidischen Distanzfunktion verwendet. Dieser nichtparametrische Klassifikator bietet den Vorteil, dass er mit einer großen Zahl an Klassen gut umgehen kann, ein Overfitting der Parameter vermieden wird (was bei lernbasierten Verfahren ein zentrales Problem darstellt) und keine Lern- bzw. Trainingsphase nötig ist. Die Stichprobe wurde durch mehrere kd-Bäume indiziert, worauf eine approximative nächste Nachbarn (ANN) Suche angewendet wurde.

Zwar ist das Ziel von [AF10] nicht direkt die Erkennung von allgemeinen Objekten, sondern nur von Wahrzeichen, das vorgestellte Verfahren ist dennoch interessant, da es ähnlich zu [BSI08] erweiterbar ist. Die Bilder werden durch SIFT- und Speeded-Up Robust Features (SURF)-Deskriptoren beschrieben, jedoch werden die Deskriptoren nicht durch ein visuelles Vokabular quantisiert. Es werden mehrere verschiedene NN-Klassifikatoren z. T. aufbauend auf Ansätzen von [Low04] vorgestellt und evaluiert. Details zu den verschiedenen NN-Klassifikatoren sind in [AF10] zu finden.

4.1.6.3 Generative Ansätze

In den PASCAL Wettbewerben und bei Messungen auf den Caltech Datensätzen schnitten diskriminative Ansätze am besten ab. Zur Vollständigkeit werden einige generative Verfahren kurz erläutert. In [WZFF06] wird ein generatives Verfahren aufbauend auf dem HDP von [TJBB06] eingesetzt, welches jedoch in Relation der Anzahl von Trainingsbildern und den erreichten Klassifikationsraten auf dem Caltech 101 Datensatz schlechter abschneidet als z. B. der Spatial-Pyramid-Ansatz von [LSP06]. In [SRE⁺05] wurde gezeigt, dass pLSA durchaus geeignet zum Erlernen von unterschiedlichen Klassen ist. In [BZM08] wurde diese Erkenntnis bzgl. pLSA in Kombination mit diskriminativen SVMs und der k-nächste Nachbarn (kNN) Suche evaluiert. Später, in [BR07] wurde die Verwendung von pLSA jedoch verworfen, da die zusätzliche Berechnung von pLSA die Berechnungszeit erheblich erhöht und dabei nur wenig zur Verbesserung der Klassifikationsrate beiträgt.

4.1.6.4 Zusammenfassung

Ein Vergleich der vorgestellten Objekterkennungsansätze ist in den Tabellen 4.7 und 4.8 zusammengefasst.

In diesem Abschnitt wurden die besten aktuellen Verfahren zur allgemeinen Objekterkennung in Bildern kurz vorgestellt. Der Vergleich der verschiedenen Ansätze zeigt, dass die Verbindung von sparse und dense Sampling zur Ermittlung von interessanten Punkten vorteilhaft ist. Zur Beschreibung der Punkte wird bei jedem Verfahren der SIFT- oder ein SIFT-ähnlicher Deskriptor verwendet. Bei den Wettbewerben schnitten Ansätze mit diskriminativen Klassifikationsmethoden am besten ab. Als Klassifikator wurden meistens SVMs verwendet, wobei am häufigsten die Spatial-Pyramid-Kernfunktion nach [LSP06] in Verbindung mit der χ^2 -Distanz eingesetzt wurde. Die Erweiterbarkeit durch neue Objektklassen und die Skalierbarkeit auf mehrere 100 oder 1 000 Objektklassen wurde kaum berücksichtigt. Lediglich die Verfahren [BSI08] und [AF10] konnten im Ansatz als erweiterbar identifiziert werden, obwohl die Erweiterbarkeit selber nicht angesprochen wurde. Die genauen Probleme bzgl. der Erweiterbarkeit von bisherigen Ansätzen werden in Abschnitt 5.1 analysiert.

Ansätze zur hierarchischen skalierenden Objekterkennung werden im folgenden Abschnitt kurz vorgestellt.

GEN. / DISK.	DETEKTOR	DESKRIPTOR	RÄUMLICHE BEZ.	DISTANZMASS	KLASSIFIKATION	DATENSATZ	EVALUATION
[ZMLS07]	-/√	Harris-Laplace, Laplacian, Harris-Affine	SIFT (SPIN,RIFT)	EMD, χ^2	SVM	XeroX7, Caltech 6,101, Graz, PASCAL	ROC, Ausführungszeit
[MSHW07]	-/√	Harris-Laplace, Laplacian, dense sampling	SIFT, SIFT, PAS	Grid (1x1,2x2,3x1) χ^2	multichannel Gaussian Kernel SVM, genetischer Algorithmus	PASCAL VOC 2007	Precision, Recall
[GMS08]	-/√	Harris-Laplace, Laplacian, Hessian, dense sampling	SIFT, Op-ponentSIFT	Grid (1x1,2x2,3x1) χ^2	SVM multichannel RBF Kernel, random-restart hill climbing	PASCAL VOC 2008	Precision, AP
[TSU+08]	-/√	Harris-Laplace, dense sampling	SIFT, Op-ponentSIFT, rgSIFT, WSHIFT, transformed color SIFT	Grid (1x1,2x2,3x1) χ^2	SRKDAA	PASCAL VOC 2008	AP
[KS07]	-/√	dense sampling	Farbhistogramm, SIFT, PCA-SIFT	Grid (1x1,2x2,3x3,4x4)	SKM	Caltech 101,256	AP
[VR07]	-/√	dense sampling	SIFT, SIFT, PHOG, GB	Grid (1x1,2x2)	SVM	Caltech 101	AP

Tabelle 4.7: Vergleich von den besten Objekterkennungsansätzen – Teil 1.

	GEN./ DISK.	DETEKTOR	DESKRIPTOR	RÄUMLICHE BEZ.	DISTANZMASS	KLASSIFIKATION	DATENSATZ	EVALUATION
[BR07]	-/-	Canny+Sobel, dense sampling	PHOG, GIST, GB, Self- Similarity	ROI Grid (1x1,2x2,4x4)	χ^2	SVM	Caltech 101,256	AP
[Per08]	/\ /	dense sampling	SIFT, RGB- Statistiken	Grid (k. A.)	L_2	NN	Bhattacharyya Sparse Regression (durch berechnet)	F-Measure, AUC, Aus- führungszeit
[BSI08]	-/\ /	dense sampling	Farben, SIFT, Self- Similarity				Xerox7, Logi- c Regression [OT01], diverse PASCAL VOC 2006	AP
[BZM08]	\ / \ /	Harris, dense sampling	SIFT, HSV- SIFT, Farb- histogramm, HoG	Grid (1x1,2x2,4x4)	χ^2	SVM, pLSA (durch EM berechnet)	kNN, [OT01], diverse Caltech 101,256	AP
[WZFF06]	\ / -	Saliency	gradient bin (SIFT- ähnlich), anschl. PCA	Linkage	HDP MCMC (durch berech- net)	(durch 4,101)	Caltech Confusion Matrix	
[SRE ⁺ 05]	\ / -	Harris-Affine, MSER	SIFT	doublet (nur Gesichtern getestet)	Kullback- Leibler	pLSA (durch EM berechnet)	MIT, Caltech 101	ROC, Confu- sion Matrix

Tabelle 4.8: Vergleich von den besten Objekterkennungssätzen – Teil 2.

4.1.7 Hierarchien für die Objekterkennung

Die Einführung von Hierarchien in den Prozess der Objekterkennung bringt zwei wesentliche Vorteile mit sich: Zum Einen kann durch die Baum- oder Graphstruktur der Suchraum bei der Klassifikation bei jedem Schritt erheblich verkleinert werden. Hierdurch wird der Aufwand des üblichen SVM-Ansatzes mit linearer Komplexität reduziert. Andererseits können in die Hierarchien auch semantische Beziehungen zwischen Objekten integriert werden (z. B. aus WordNet [Mil95]), wodurch die Genauigkeit verbessert werden kann. Die in WordNet verwendeten linguistischen Beziehungen zwischen Wörtern werden in Anhang B beschrieben.

In diesem Abschnitt werden Ansätze zur Integration von Hierarchien in die Objekterkennung vorgestellt.

4.1.7.1 Textueller Vergleich

[Han07] hat die in Datensätzen verwendeten Begriffe, welche häufig zur Evaluierung von Objekterkennungsansätzen verwendet wurden, mit den Konzepten von WordNet verglichen. Die Untersuchung hat gezeigt, dass in aktuellen Datensätzen hauptsächlich die Erkennung von Objekten und Lebewesen berücksichtigt wurde. Es existieren viele Zweige in WordNet, welche durch diese Datensätze nur gering abgedeckt sind. WordNet scheint damit also nicht die intuitivste Worthierarchie für die Annotation von Bildern bereitzustellen. Ein Vergleich mit anderen Konzept-Thesauri (siehe Abschnitt 3.3.2) wurde nicht unternommen.

4.1.7.2 Strukturierung von interessanten Punkten

[OM05] stellt einen binären Entscheidungsbaum zur hierarchischen Strukturierung von interessanten Punkten in Bildern vor. Jeder innere Knoten teilt basierend auf einfachen Pixelwerten den Baum auf zwei Teilbäume auf. Die Ermittlung der interessanten Punkte geschieht mit dem MSER-Detektor. Das Erlernen des Baumes muss offline durchgeführt werden. Außerdem muss der Baum auch neu erstellt werden, sofern ein neues Objekt mit aufgenommen werden soll. Zur Klassifikation eines neuen Bildes werden zuerst die interessanten Punkte extrahiert, anschließend mittels des Entscheidungsbaumes klassifiziert und die erhaltenen Blattknoten sequentiell auf Ähnlichkeit geprüft. Der Ansatz wurde auf zwei Datensätzen getestet und hat dabei gute Ergebnisse erzielt.

In [NS06] wird eine Idee zur Organisation der visuellen Wörter vorgestellt. Bei dieser Vorgehensweise wird für die Quantisierung der visuellen Wörter nicht das übliche k -means Verfahren eingesetzt, sondern es wird eine hierarchische Baumstruktur durch ein top-down hierarchisches k -means Clustering aufgebaut, wobei k hier für den Verzweigungsgrad des Baumes steht. Für die zu erreichende maximale Höhe des Baumes wird der Parameter L verwendet. An den Blättern des Vokabularbaumes werden mittels einer Scoring Funktion invertierte Listen der relevantesten Bilder erstellt. Für die inneren Knoten werden diese Listen aus den invertierten Listen der Blattknoten berechnet. Der erstellte Vokabularbaum bleibt anschließend unverändert, es werden lediglich Bilder in die invertierten Listen eingefügt.

Abbildung 4.15 zeigt das Vorgehen aus [NS06]. Der Ansatz wurde mit mehreren Datenbanken unterschiedlicher Größe (6 376, 40 000 und 1 000 000 Bilder) getestet und ausgewertet. Die Evaluation hat gezeigt, dass der Ansatz umso besser funktioniert, je mehr Blattknoten vorhanden sind.

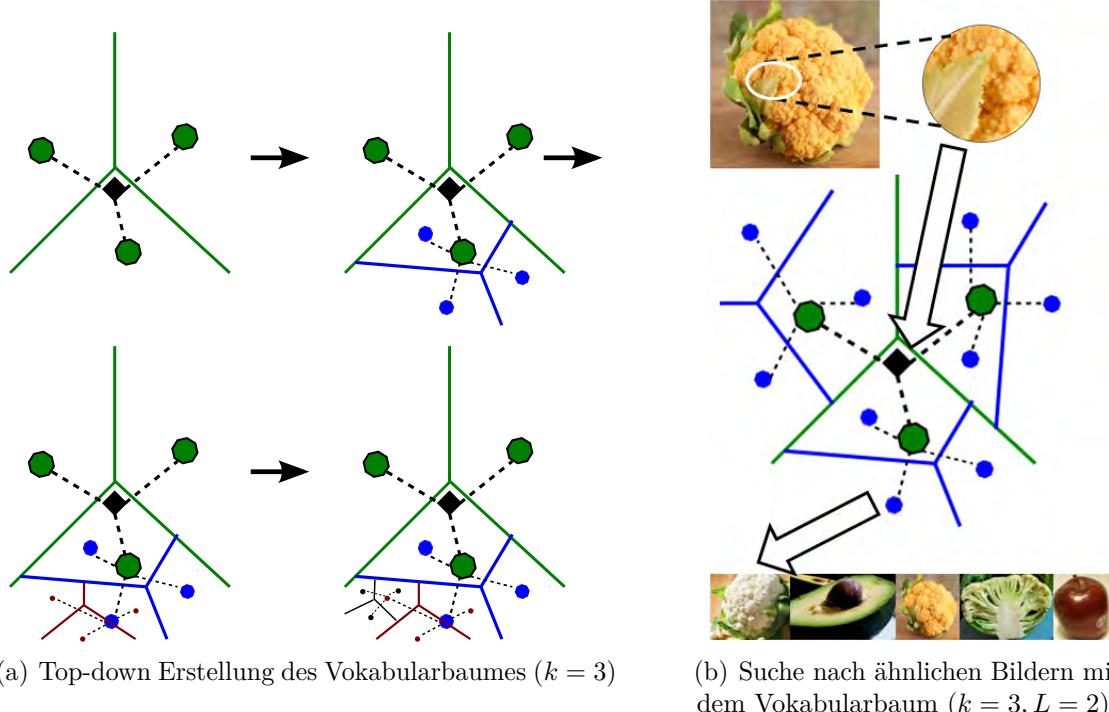


Bild 4.15: Objekterkennung mit einem Vokabularbaum nach [NS06].

In [PCI⁺07] wird das hierarchische k -means Verfahren zur Erstellung des Vokabulars aus [NS06] mit einem approximativen und dem einfachen k -means Algorithmus verglichen. Als Bewertungsmaß wurde MAP verwendet. Die Auswertungen mit Vokabularen der

Größenordnung von 1 000 000 visuellen Wörtern haben gezeigt, dass das approximative Verfahren wesentlich besser abschnitt als der hierarchische Ansatz.

4.1.7.3 Klassifikationshierarchien

Klassifikationsprobleme, bei denen mehrere Klassen von einander getrennt werden sollen, werden häufig durch die Kombination von mehreren binären Klassifikatoren realisiert. In der allgemeinen Objekterkennung werden oft SVMs mit verschiedenen Kernfunktionen eingesetzt. Dabei werden binäre SVMs auf folgende Weisen kombiniert:

- Wettbewerb [RK04] (Einer gegen Rest, SVM 1:N)
- Abstimmung (Einer gegen Einen, SVM 1:1)
- sukzessiver Ausschluss nach [PCST00] (gerichteter azyklischer Graph, Directed Acyclic Graph (DAG) SVM)

Die verschiedenen Kombinationsmöglichkeiten sind in Abbildung 4.16 beispielhaft dargestellt. Es ist offensichtlich, dass die Komplexität bei der 1:1-Methode quadratisch, bei der 1:N-Lösung und beim gerichteten azyklischen Graphen linear von der Anzahl der Klassen abhängt.

Nach der Schätzung von [Bie87] kann ein Mensch aus psychologischer Sicht ca. 30 000 Objekte schnell unterscheiden. Da die allgemeine Objekterkennung die Unterscheidung einer ähnlich hohen Anzahl an Objektklassen als Ziel verfolgt, besteht Bedarf an Lösungen mit polylogarithmischer, logarithmischer oder konstanter Komplexität.

In diesem Abschnitt werden Ansätze zur Lösung dieses Problems vorgestellt.

In [LYCR05] wird ein binärer SVM-Baum zur adaptiven hierarchischen Klassifikation von Bildern vorgestellt. Zur Ermittlung der Teilbäume wird für jeden Knoten ein k -means Clustering Algorithmus mit $k = 2$ durchgeführt. Für jeden inneren Knoten wird ein SVM-Klassifikator trainiert. Je nach Trainingsmenge kann der erlernte Binärbaum eine mehr oder weniger entartete Struktur aufweisen. Im Extremfall ist der Binärbaum genau so langsam wie eine lineare Kette von SVM-Klassifikatoren (1:N SVMs). Ein weiteres Problem ist auch, dass der Baum bei der Hinzunahme einer neuen Objektklasse komplett neu berechnet werden muss.

In [YLM⁺06] wird ein ähnlicher Ansatz zur Klassifikation von Videos nach deren Genre verfolgt. Dieses Verfahren weist ähnliche Probleme auf wie [LYCR05].

[CW05] realisiert einen balancierten binären SVM-Baum.

[CCG04] verwendet ebenfalls binäre hierarchische SVMs, wobei hier für die Aufteilungen

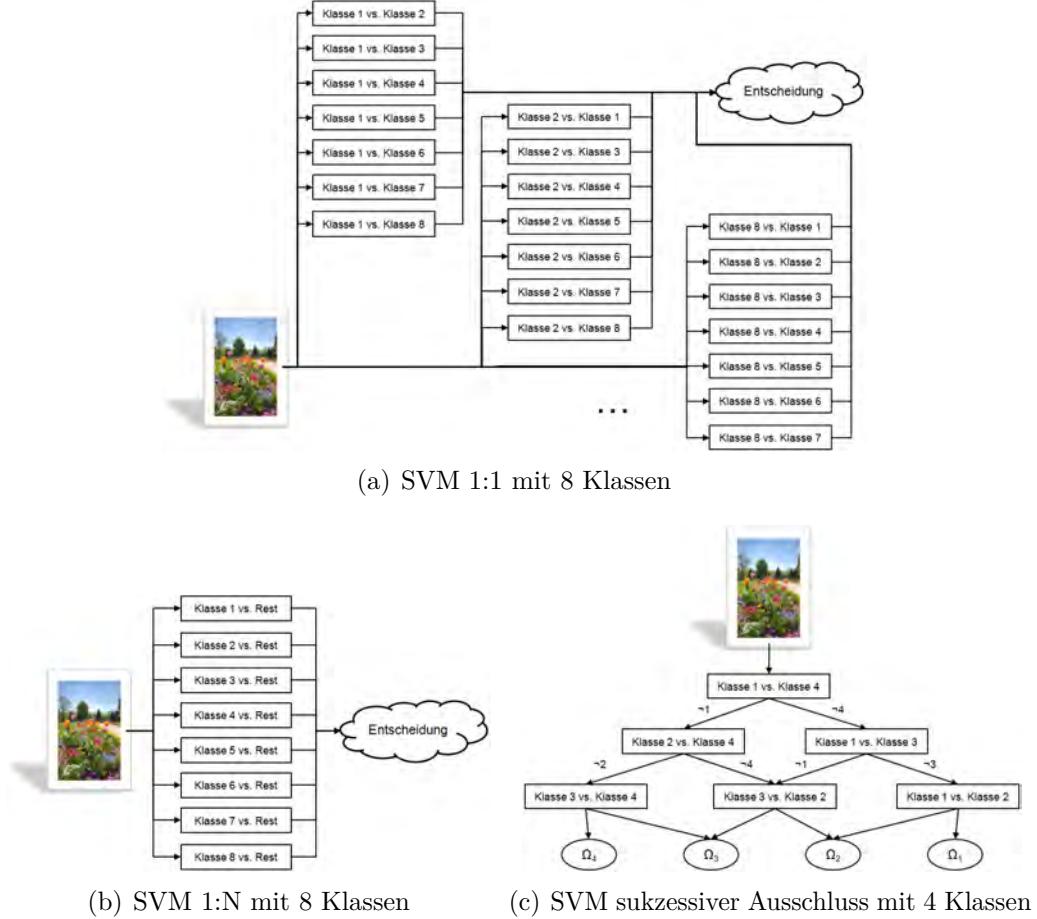


Bild 4.16: Kombinationsmöglichkeiten von SVMs für die Klassifikation in mehrere Klassen.

der Trainingsmenge bei den inneren Knoten die Kullback-Leibler-Divergenz verwendet wird.

In [ZWQ⁺05] werden neben binären auch k -näre hierarchische SVMs vorgestellt. Beide wurden als ähnlich genau, jedoch schneller befunden als traditionelle Multiklassen-SVM-Ansätze.

In [ZW07] wurde überprüft, welche Auswirkungen die Kombination von Klassifikatoren der inneren Knoten einer natürlichen Objekthierarchie mit klassenspezifischen Klassifikatoren hat. Innere Knoten liefern einen höheren Recall-Wert, während klassenspezifische Klassifikatoren eine höhere Precision haben. Die Untersuchungen haben gezeigt, dass mit dem vorgestellten Kombinationsansatz sowohl der höhere Recall als auch die höhere Precision beibehalten werden kann. Der Ansatz wurde mit vier verschiedenen natürlichen Objekthierarchien evaluiert.

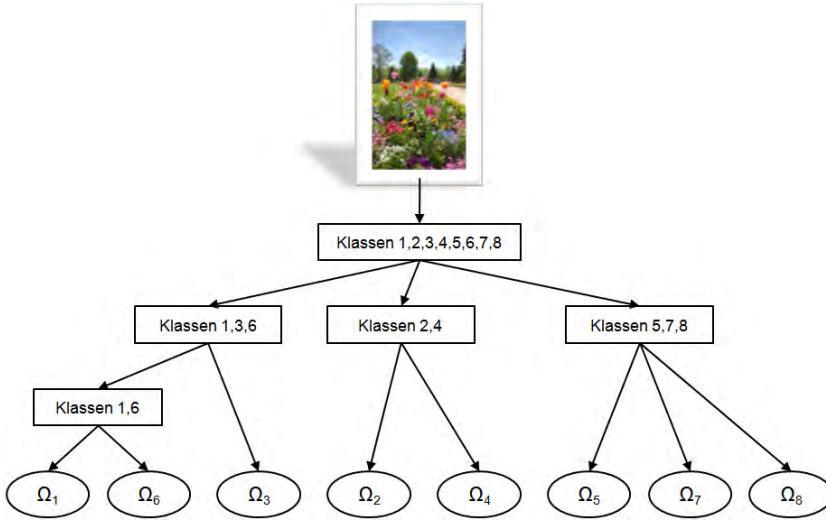


Bild 4.17: k -närer SVM-Baum nach [ZWQ⁺05] mit 8 Klassen.

[MS07] erweitert die Labels der zum Training benutzten Bilder mit Homonym-, Synonym-, Meronym- und Holonymbeziehungen aus WordNet (vgl. Anhang B). In Verbindung mit dem Ansatz aus [ZMLS07] wird ein durch Wortbeziehungen erweitertes hierarchisches Modell mittels SVMs erlernt. Das Verfahren verhielt sich bei der Evaluation auf dem PASCAL VOC 2006 Datensatz bzgl. der Genauigkeit ähnlich wie die besten Ansätze aus dem Jahr 2006. Ein wesentlicher Unterschied ist jedoch, dass es unterhalb der linearen Komplexität von 1:N SVMs blieb.

In [BPPW08] wird der Nested Chinese Restaurant Process (NCRP) Ansatz aus [BGJT04] für die Erstellung einer hierarchischen visuellen Taxonomie mittels visuellen Wörtern adaptiert. In einer Untersuchung mit 13 Klassen wurde festgestellt, dass das Verfahren auch ohne der Berücksichtigung von Klassenlabels eine Taxonomie erstellen kann, welche bei der Klassifikation zu 58% korrekt ist. Unter Verwendung von Klassenlabels wurde der Ansatz mit dem LDA-basierten Verfahren aus [FFP05] verglichen und bzgl. der Klassifikationsrate als besser befunden.

[GP08] erweitert den Spatial-Pyramid-Ansatz von [LSP06] um eine aus Trainingsdaten erlernte hierarchische Taxonomie basierend auf dem Caltech 256 Datensatz. Die Taxonomie ist ein Binärbaum, bei dem die schwierigsten Entscheidungen zwischen Klassen möglichst tief in den Baum geschoben werden. Um dies zu erreichen, wird eine Confusion Matrix (siehe Abschnitt 4.1.4) mit dem Verfahren von [LSP06] aus den Trainingsdaten berechnet. Für das Erlernen einer Taxonomie werden zwei Ansätze verglichen. Für den top-down Ansatz wird der Spectral-Clustering-Algorithmus von [ZMP04] verwendet. Der

bottom-up Ansatz verbindet schrittweise die Klassenpaare mit der größten Verwechslung und aktualisiert nach der Zusammenlegung die Zeilen und Spalten der Confusion Matrix mit Durchschnittswerten. Die so erhaltenen Taxonomien weisen beide eine ähnliche Klassifikationsrate auf. In der erlernten Taxonomie werden für alle inneren Knoten des Binärbaums SVMs mit der Spatial-Pyramid-Kernfunktion trainiert, wozu jeweils 10% der Teilbäume als Trainingsdaten verwendet werden. Bei der Analyse der erlernten Taxonomie wird – ähnlich wie in [Han07] auch hier – angemerkt, dass lexikalische Beziehungen zwischen Kategorien (z. B. entnommen aus WordNet) von visuellen Beziehungen abweichend sein können.

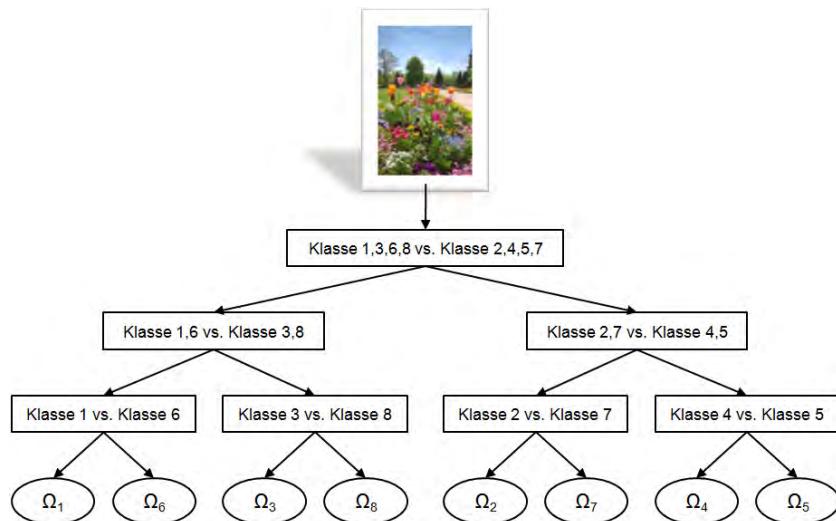


Bild 4.18: Binärer SVM-Baum nach [GP08] mit 8 Klassen.

In [MS08] werden bisherige top-down und bottom-up Ansätze hinterfragt, inwieweit die strikte Trennung der einzelnen Pfade in einem Klassenbaum sinnvoll ist. Hauptaugenmerk war dabei, dass bisherige Verfahren in ihren entsprechenden Veröffentlichungen nur mit 10 bis 14 Klassen evaluiert wurden und diese mit weit mehr Klassen nicht skalieren. In [MS08] wurde eine Abwandlung des DAG-SVMs von [PCST00] vorgestellt, welches das Erreichen eines Blattknotens (Klasse) über mehrere Pfade ermöglicht. Der wesentliche Unterschied zum Ansatz aus [PCST00] ist, dass bei den inneren Knoten nach Trainingsdaten aufgeteilt wird und die Überlappung durch Parameter eingestellt werden kann. Bei der Aufteilung wird die Stichprobe ω auf disjunkte Mengen ω_A und ω_B und dabei die Klassenmenge Ω auf Klassenmengen Ω_A , Ω_B und Ω_X gesplittet. Der Klassenmenge Ω_A sind alle Stichproben der Menge ω_A zugeordnet (analog Ω_B und ω_B). Ω_X beinhaltet diejenigen Stichproben, die sowohl zur Menge ω_A als auch zur Menge ω_B gehören. Der linke Teilbaum besteht somit aus den Klassenmengen Ω_A und Ω_X , während der rechte

Teilbaum die Klassenmengen Ω_B und Ω_X enthält. Aus dieser Struktur folgt, dass die Blattknoten auf mehreren Pfaden erreicht werden können. Die Überlappung der Teilbäume kann durch einen Parameter α eingestellt werden.

Zur Erstellung der Histogramme von visuellen Wörtern wurde ein Vokabular mit 8 000 Wörtern verwendet. Für die inneren Knoten wurden jeweils SVMs mit einer erweiterten Gauß-Kernfunktion und unter Verwendung der χ^2 Distanz trainiert. Messungen auf dem Caltech 256 Datensatz haben gezeigt, dass bei der Verwendung gleicher Kanäle dieses Verfahren im Vergleich zu den bisherigen Ansätzen besser skaliert. Ein weiterer Vorteil bei diesem Ansatz ist, dass durch einen Parameter der Kompromiss zwischen Berechnungsgeschwindigkeit und Genauigkeit eingestellt werden kann.

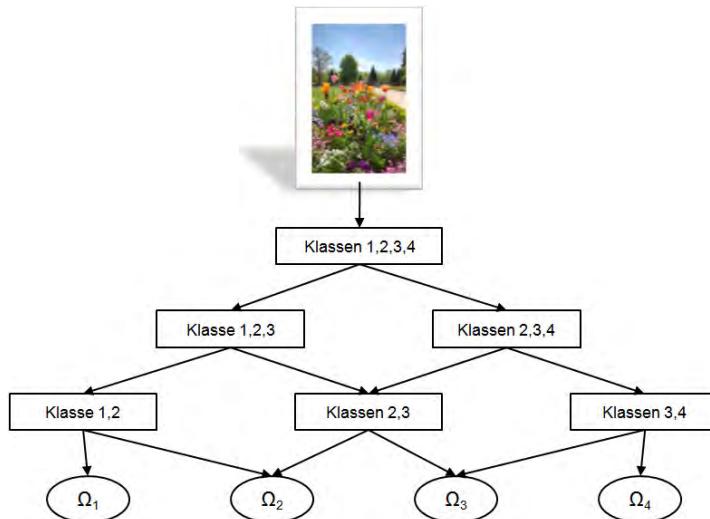


Bild 4.19: DAG-SVM nach [MS08] mit 4 Klassen.

4.1.7.4 Zusammenfassung

In diesem Abschnitt wurden Verfahren zur hierarchischen Objekterkennung erläutert. Die meisten Ansätze verfolgen dabei das Ziel, Berechnungszeit für die Klassifikation einzusparen und dabei die allgemeine Objekterkennung auf mehrere 100 oder 1 000 Objektklassen zu skalieren. Die vorgestellten Verfahren sind nur erste Schritte auf dem Gebiet der Skalierbarkeit. Die Erweiterbarkeit durch die Hinzunahme von weiteren Objektklassen während der Laufzeit wurde in keinem der Ansätze behandelt.

4.1.8 Zusammenfassung

In diesem Abschnitt wurde das Bag-of-Words-Konzept und der gesamte Ablauf der Objekterkennung in Bildern vorgestellt. Anschließend wurden die besten Verfahren aus der Literatur kurz erläutert.

Für die Ermittlung der interessanten Punkte kann festgestellt werden, dass die Verbindung von Blob- und Ecken-ähnlichen Detektoren in Kombination mit einem festen Gitternetz die besten Resultate erzielt. Diese Punkte werden in den meisten Ansätzen durch SIFT- oder SIFT-ähnliche Deskriptoren beschrieben. Sowohl die Hinzunahme von Farbinformationen als auch die Verwendung von Formdeskriptoren wie z. B. PHOG oder PAS hat die Erkennungsrate verbessert. Zur Ermittlung des visuellen Vokabulars wird der k -means Clustering-Algorithmus eingesetzt. Für die Beschreibung der räumlichen Zusammenhänge zwischen den interessanten Punkten hat sich der Spatial-Pyramid-Ansatz von [LSP06] etabliert. Das Distanzmaß variiert zwar, es wird allerdings häufig χ^2 verwendet. Durch die Kombination verschiedener Deskriptoren, Detektoren und Grids ergeben sich vielfältige „Kanäle“. Die Gewichtung dieser Kanäle für den Klassifikator wird unterschiedlich erlernt. Als Klassifikatoren werden bei diskriminativen Verfahren fast ausschließlich SVMs mit erweiterten Gauß- bzw. Spatial-Pyramid-Kernfunktionen eingesetzt, einige wenige Ansätze verwenden jedoch NN-Klassifikatoren mit ähnlich guten Erkennungsraten. Bei generativen Verfahren ist die Vielfalt bei Klassifikatoren erheblich größer. Hier kommen Ansätze wie z. B. pLSA oder HDP zum Einsatz.

Kaum untersucht ist die Erweiterbarkeit der Verfahren durch später hinzukommende neue Objektklassen. Ein weiterer Aspekt ist auch die Hierarchisierung der Objektklassen zur Erhöhung der Genauigkeit der Klassifikation und zur Einsparung von Berechnungszeit. [Per08] und [BR07] erwähnen zwar die Möglichkeit zum Aufbau einer Hierarchie, untersucht bzw. realisiert wurde es jedoch nicht. In Abschnitt 4.1.7.3 wurden die ersten Schritte in die Richtung von hierarchischer Objekterkennung erläutert. Weiteres Optimierungspotenzial besteht auch noch bei der Bestimmung von relevanten Kanälen bzw. Merkmalen. Zur Zeit gehen die Ansätze in die Richtung so viele Kanäle und Merkmale wie möglich zu extrahieren, um diese dann durch verschiedene Lernverfahren für den Klassifikator möglichst optimal zu gewichten.

4.2 Aktuelle Annotationsverfahren für Bilder

Die Annotationsansätze werden in diesem Abschnitt bzgl. dem Beteiligungsgrad des Menschen untersucht. Dabei können manuelle, semi-automatische und vollautomatische Annotationsverfahren unterschieden werden. In [KB96] werden semi-automatische Verfahren auch als hybrid bezeichnet. Die Trennung zwischen semi- und vollautomatischer Annotation in dieser Arbeit orientiert sich an [KB96] und [FBY92] und wird in diesem Abschnitt wie folgt verwendet: semi-automatische Verfahren erfordern bei der Annotation der Bilder direkt manuellen Eingriff, während bei der vollautomatischen Annotation die Beschreibung der Bilder komplett selbstständig erfolgt, ein manueller Eingriff jedoch beim Training während dem Aufsetzen des Systems erlaubt ist. Diese Trennung basiert auch auf der Wahrnehmung des Menschen, dem am Anfang seines Lebens ebenfalls erklärt werden muss, welches Objekt was darstellt, bevor er sich dann später selbstständig – „vollautomatisch“ – orientieren kann.

4.2.1 Manuelle Annotation

Bildannotationen manuell zu erstellen ist zwar die einfachste Möglichkeit, jedoch ist sie mit zunehmender Anzahl von zu annotierenden Bildern nicht realisierbar. Je nach den gegebenen Freiheitsgraden der indizierenden Personen können die Beschreibungen variieren. In [MW03] werden hierfür folgende Gruppen beschrieben:

- Zuordnung von frei wählbaren Schlag- oder Stichwörtern,
- Wissensrepräsentationstechniken (Prädikate, semantische Netze, Frames, Scripts, etc.),
- Zuordnung von Schlagwörtern entnommen aus einem Thesaurus,
- Attribut-Wert-Paare
- freier Text,
- Schlagzeilen, Telegrammstil.

Jede dieser Annotationsmöglichkeiten hat neben dem Problem, dass es mit hohem Zeit- und Kostenaufwand verbunden ist, auch den Nachteil, dass die erzeugten Beschreibungen, sofern sie von mehreren Personen erstellt wurden, durch den unbestimmbaren Faktor

Mensch Inkonsistenzen und Mängel bzgl. der Zuverlässigkeit der Annotationen aufweisen können.

Dennoch ist die manuelle Annotation bzw. das Tagging recht verbreitet, vor allem wegen des Social Webs. In Flickr können z. B. alle registrierten Benutzer beliebige Bilder von anderen mit weiteren Tags versehen. Weitere Bedeutung im Webumfeld ist auch den `alt-` und `longdesc-`Tags bei Bildern in HTML zuzuschreiben, in denen der Ersteller der Webseite beliebige alternative Beschreibungen für das Bild angeben kann. Sowohl das Tagging als auch die manuelle Beschreibung über `alt-`Tags wirken sich positiv auf die Suche nach multimedialen Inhalten aus, da zur Zeit große Suchmaschinen wie Google oder Yahoo! noch keine Indizierung basierend auf dem Bildinhalt durchführen. Daraus folgt auch, dass das Suchergebnis sehr stark von der Güte der Annotation abhängt. Es ist klar ersichtlich, dass eine effiziente vollautomatische inhaltsbasierte Annotation und Indizierung von Bildern signifikant die Kosten und den Zeitbedarf senken sowie von wesentlich einheitlicherer Qualität sein würde.

Die ersten Schritte zur Unterstützung der annotierenden Person wurden im wohl allerersten System, welches anhand von den tatsächlichen Bildinhalten nach Bildern suchen konnte, unternommen. Die Annotationswerkzeuge für das Query by Image Content (QBIC) System werden in [FSN⁺95] beschrieben. Unter anderem können die Werkzeuge bei einfachen Bildern das Objekt vom Hintergrund trennen, ein Objekt mit einem Flächenbrand-ähnlichen Ansatz markieren sowie die vom Benutzer eingezeichnete Umrandung von Objekten den Rändern des Objekts genauer anpassen. Die textuelle Annotation erfolgt vollständig manuell.

In [HSWW03] wird ein Ansatz beschrieben, mit dem die manuelle semantische Annotation von Bildern durch verschiedene Ontologien unterstützt werden kann. Verwendete Thesauri bzw. Ontologien sind: WordNet¹ [Mil95, Fel98], Art & Architecture Thesaurus, Iconclass und Union List of Artist Names. Es wurde auch ein manuelles Annotationswerkzeug erstellt, bei dem vom Resource Description Framework (RDF) ausgiebig Gebrauch gemacht wird.

Eine einfache Hilfe zur manuellen regionsbezogenen Annotation von Bildern wird in [RTMF08] vorgestellt. Das webbasierte Werkzeug LabelMe² (siehe Abbildung 4.20) ermöglicht es den Benutzern, beliebige Regionen im Bild durch Polygone zu markieren

1 <http://wordnet.princeton.edu/>

2 <http://labelme.csail.mit.edu/>

und mit Beschreibungen zu versehen. In [RTMF08] werden die mittels LabelMe gesammelten Daten auch aus verschiedenen Blickwinkeln ausgewertet. Erwähnenswert ist auch die Einbeziehung von WordNet zur Bereinigung der von den Benutzern verwendeten verschiedenen Begriffe.

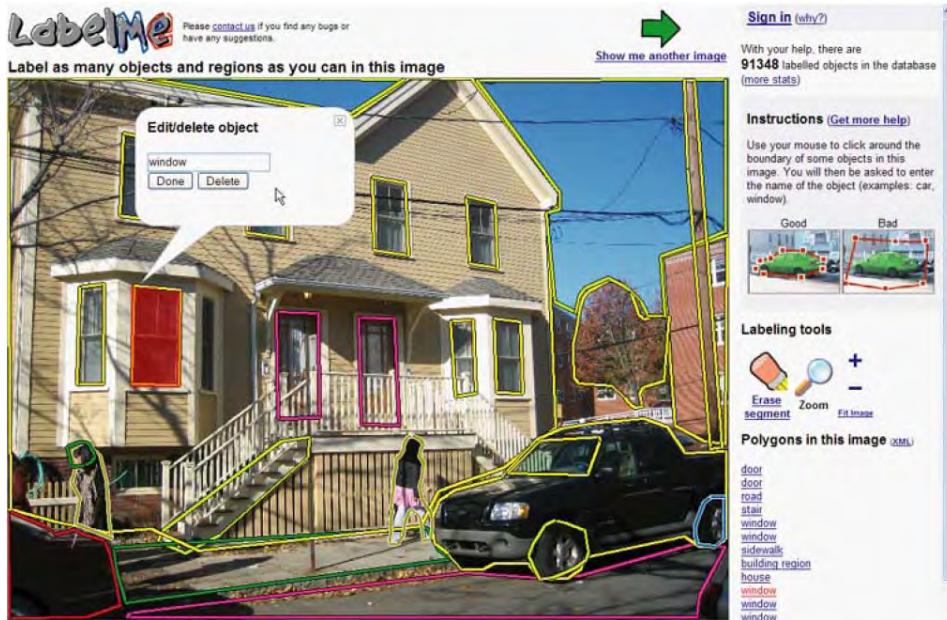


Bild 4.20: LabelMe: ein webbasiertes Werkzeug zur manuellen Annotation von Bildern.

Um Webnutzer zur Annotation von Bildern zu motivieren, wurde in [AD04] ein Extra Sensory Perception (ESP) Spiel vorgestellt. Dabei werden unterschiedlichen Spielern gleichzeitig identische Bilder gezeigt. Die Spieler müssen, ohne zu wissen, was der andere Mitspieler schreibt, erraten, wie der Gegenspieler das Bild beschreiben würde. Bei einem Treffer erscheint das nächste Bild und Punkte werden für die Anzahl der benötigten Versuche verteilt. Nachteil beim ESP-Spiel war, dass bereits ein einziges Wort für eine Übereinstimmung ausreichend war, was jedoch recht dürfsig für die Nutzung der Beschreibung der Bilder ist.

Das Spiel wurde in [AGK⁺06] zu einem Frage-Antwort-Spiel weiterentwickelt. Hier muss ein Spieler ein Bild möglichst genau beschreiben, während die anderen Spieler das zur Beschreibung passende Bild mittels einer bereitgestellten Suchmaschine auffinden müssen. Somit können Bilder spielerisch sehr gut durch Freitext annotiert werden. Die so erhaltenen Annotationen entsprechen auch in den meisten Fällen den Anforderungen von Blinden aus [PHD05] (siehe Abschnitt 3.1.2).

4.2.2 Semi-automatische Annotation

Die hybride bzw. semi-automatische Annotation wird in [KB96] beschrieben als die automatische Ermittlung von einigen oder allen Wörtern, die im Anschluss durch die annotierende Person korrigiert oder ergänzt werden müssen. Hierzu zählen also sowohl Werkzeuge, die automatisch ermittelte Empfehlungen zur Annotation geben, als auch Relevance-Feedback-Mechanismen, bei denen der Benutzer das Ergebnis interaktiv bewerten und somit zukünftige Suchergebnisse beeinflussen kann.

In [WDS⁺01] wird eine semi-automatische Annotation von Bildern vorgestellt, die auf Relevance Feedback basiert. Der Benutzer erhält als Grundlage eine Menge von Bildern, die mehr oder weniger gut annotiert sind, und in der er sowohl text- als auch inhaltsbasiert suchen kann. Bei einer textbasierten Suche werden mittels inhaltsbasierter Ähnlichkeiten zu anderen Suchergebnissen z. T. auch Bilder zurückgeliefert, die nicht mit dem Suchbegriff annotiert wurden. Sofern der Benutzer diese Bilder als relevant einstuft, wird der Suchbegriff in die Annotation mit aufgenommen.

In [MM04] wird versucht, mittels Inferenznetzwerken die Anfragen an Bildretrievalssysteme genauer zu formulieren. Die vorgeschlagene Vorgehensweise erlaubt auch das Verbinden von textuellen und visuellen Anfragen mittels boolescher Operatoren. Gerade durch diese Verknüpfung können ggf. auch Bilder gefunden werden, die visuell ähnlich sind, jedoch zur Zeit noch nicht textuell annotiert wurden. Hiermit öffnet sich auch das Potenzial, den Anfragemechanismus ähnlich wie in [WDS⁺01] zur Bildannotation zu missbrauchen.

In [DI03] wird ein semiautomatisches Annotationsverfahren vorgestellt, welches eine bereits annotierte Bilddatenbank als Grundlage benötigt. Bei neuen Bildern wird zuerst eine Farbsignatur ermittelt und anschließend mittels der Earth-Mover Distanz mit den Bildern in der Datenbank verglichen. Im nächsten Schritt werden die Annotationen der ähnlichsten Bilder mittels Frequent Keyword Mining untersucht, wodurch eine Annotation für das zu annotierende Bild ermittelt wird. Die annotierende Person kann im Anschluss diese Menge von Schlüsselwörtern korrigieren, einschränken oder ergänzen.

In [CJS05] wird der Ansatz zur automatischen Ermittlung von Konzept-Hierarchien aus der Domäne des Text-Retrievals auf Bilder angewendet. Hierzu werden ausschließlich die textuellen Beschreibungen der Bilder benutzt und der Algorithmus identisch wie beim Text-Retrieval für Dokumente umgesetzt. Zur Erstellung der Konzept-Hierarchie werden die Terme der Top N Bilder zu einer gegebenen textuellen Anfrage auf Häufigkeiten untersucht. Außerdem wird versucht zusammenfassende Terme, Oberbegriffe statistisch

zu ermitteln, wodurch eine Hierarchie entsteht. Zu jedem Term auf jeder Hierarchiestufe wird ein Beispielbild und die Anzahl der untergeordneten Bilder angezeigt. Die Auswahl der Beispielbilder erfolgt zufällig.

Nur wenige semi-automatische Systeme sind im Web verfügbar gemacht worden. Eines dieser seltenen Systeme ist das in [LW06, LW08] vorgestellte Automatic Linguistic Indexing of Pictures - Real-Time (ALIPR)¹. Ein Screenshot des semi-automatischen Annotationsdienstes ist in Abbildung 4.21 zu sehen. ALIPR ist die Erweiterung des bereits bestehenden Automatic Linguistic Indexing of Pictures (ALIP) Systems, welches nun auch Bildannotationen in nahezu Echtzeit ermöglicht. Hierzu müssen zuerst die Trainingsbilder zu den jeweiligen Konzepten manuell separiert, anschließend Merkmale extrahiert und Regionen zugeordnet werden. Danach wird ein eigenes Clusteringverfahren angewendet, um ein statistisches Modell für die jeweiligen Konzepte zu erstellen. Basierend auf diesem Modell werden zukünftige Bilder annotiert. Der Benutzer von ALIPR kann Bilder über eine Webseite hochladen und erhält eine Liste von 15 Vorschlägen, aus denen er passende Konzepte auswählen oder auch eigene Tags manuell eingeben kann. Die Qualität der Annotationen ist sehr dürftig. Leider wurde die Webseite auch mittlerweile offline genommen.

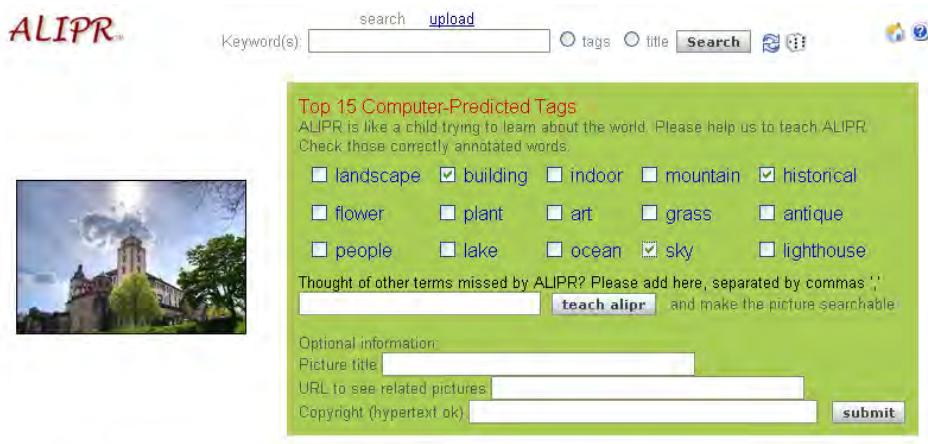


Bild 4.21: ALIPR Webseite mit Beispielbild und automatisch ermittelten Tags.

In [WZJM06a, WZJM06b] wird ein Ansatz vorgestellt, bei dem Bilder mittels der Verknüpfung von einer Suche und Data Mining annotiert werden. Der Ansatz baut auf eine bereits annotierte große Bilddatenbank auf und benötigt für das zu annotierende

1 <http://www.alipr.com>

Bild zusätzlich noch ein Schlüsselwort vom Benutzer. Mit diesem Suchbegriff werden aus der Bilddatenbank ähnliche Bilder ermittelt, die zusätzlich auch auf visuelle Ähnlichkeit untersucht werden. Anschließend werden die Ergebnisse durch ein Clusteringverfahren gruppiert und die Schlüsselwörter der Cluster als Annotation für das zu annotierende Bild verwendet. Der Ansatz hat den großen Vorteil, dass bis auf das eine Schlüsselwort alles automatisch abläuft. Die Qualität der Annotation ist allerdings abhängig von den Annotationen der verwendeten Datenbank und der richtigen Auswahl des Suchbegriffs.

4.2.3 Vollautomatische Annotation

In [FBY92] wird die automatische Extraktion einer Beschreibung als „der Prozess der algorithmischen Beobachtung von Informationseinheiten zur Generierung einer Liste von Indexterminen“ definiert. Basierend auf dieser Definition wird die vollautomatische Annotation in dieser Arbeit in Anlehnung an das Heranwachsen der menschlichen Wahrnehmung wie folgt definiert: Ein Verfahren zur Annotation gilt als vollautomatisch, sofern die Zuordnung von Text zu Bildern völlig selbstständig, d. h. ohne manuellem Eingriff erfolgt. Jedoch kann bei der Lernphase des Systems bzw. bei der Ermittlung des Grundwissens (z. B. Erstellung der annotierten Datenbank) manuelles Vorgehen eingesetzt werden.

Bereits in [MTO99] wurden die ersten Versuche zur statistischen Beschreibung des gemeinsamen Auftretens von Wörtern und Bildern mittels einem Bildraster getätigt. In [BF01] wurde dieser Ansatz zum Teil fortgesetzt, wobei hier ein hierarchisches Modell aus dem Text-Retrieval benutzt wurde. Im wesentlichen ist das Modell als ein hierarchisches Clustering-Verfahren auffassbar, bei dem von der Wurzel hinab die einzelnen Knoten immer spezifischere Wörter und Bildteile mit zugeordneten (erlernten) Wahrscheinlichkeiten darstellen. In [DBFF02] wurde dieser Ansatz erneut weitergeführt und durch Erkenntnisse aus der Textübersetzung erweitert. Die Herangehensweise beruht auf dem BoW-Ansatz, es wurde jedoch noch nicht der mittlerweile etablierte SIFT-Deskriptor verwendet. Die einzelnen Bildregionen (Bildflecken) wurden mit Normalized Cuts ermittelt und durch insgesamt 33 Merkmale beschrieben (u. a. Farbe, Position, Textur, Form). Zusätzlich wurden die Wörter basierend auf Ähnlichkeiten der Bildregionen geclustert, womit eine noch bessere Annotation angestrebt wurde. Der Ansatz aus [DBFF02] wurde in [PYDF04] und [SVBM05] direkt weitergeführt und letztendlich sind eine Vielzahl von BoW-Varianten entstanden, die in Abschnitt 4.1 vorgestellt wurden.

[BKL⁺06] orientiert sich an dem Ansatz des Mediators (Proxy) aus [KPD96, TA00, BR01], welcher die aus dem Web heruntergeladenen Bilder annotiert und anschließend an den Browser weitergibt. Der Proxy kann in diesem Ansatz durch beliebige Module erweitert werden. Initial wurden für das System drei Module implementiert:

- ein Labeling-Modul, das basierend auf dem Kontext des Bildes versucht, eine Annotation zu erstellen,
- ein OCR-Modul, das z. B. Texte aus Buttons oder als Bild dargestellten Mailadressen extrahiert,
- ein manuelles Annotationsmodul für Bilder, die von durch die Vorhergehenden nicht gut genug abgedeckt werden.

Der Ansatz wurde in [BKL⁺06] lediglich auf Buttons und Mailadressen evaluiert, wofür es auch sehr gut funktioniert hat.

In [JLM03] wird ein Cross-Media Relevance Model (CMRM) zur Annotation und zum Retrieval von Bildern vorgestellt. Basierend auf einer Trainingsmenge mit bereits annotierten Bildern wird ein statistisches Modell aufgebaut, mit dem einerseits versucht wird, Bilder automatisch zu annotieren, andererseits auch textuelle Anfragen mit aktuell noch nicht annotierten Bildern zu beantworten. In [LMJ03] wird der Ansatz um das Continuous-space Relevance Model (CRM) ergänzt, welches bedeutend bessere Ergebnisse bzgl. Recall und Precision erzielt. [FML04] verbessert die vorherigen Ansätze durch Multiple Bernoulli Relevance Modelen (MBRM). In [FML04] werden die Bilder mittels eines Gitternetzes auf Regionen unterteilt, welche im Anschluss durch 18 Farb- und 12 Texturmerkmalen beschrieben werden. Nachfolgend werden die Zuordnungen zwischen den annotierten Wörtern des Trainingsdatensatzes und den Merkmalen durch das generative MBRM-Verfahren erlernt. Der Ansatz wurde u. a. auf einem Teil des Corel-Datensatzes evaluiert und ausschließlich mit den Vorarbeiten aus [LMJ03] verglichen, wobei es bezüglich Recall und Precision leicht bessere Ergebnisse erzielt hat.

In [JM04] wird eine Vorgehensweise zur automatischen Annotation von Bildern mittels der maximalen Entropie vorgestellt. Basierend auf einem Testdatensatz bestehend aus bereits annotierten Bildern wird durch statistische Methoden die Wahrscheinlichkeit eines Schlüsselwortes für ein Bild vorausgesagt. Im wesentlichen basiert der Ansatz auf der Übersetzung von visuellen Wörtern auf textuelle Wörter.

In [FGL04] und [FGLX04] wird eine Methode vorgeschlagen, um die Semantik einer natürlichen Szene automatisch zu erkennen und zu annotieren. Hierzu werden zuerst

prägende, dominante Objekte bzw. Bereiche extrahiert und identifiziert. Dies geschieht jedoch leider nicht durch ein allgemeines Modell, sondern durch jeweils einzeln programmierte, speziell zugeschnittene Funktionen. Basierend auf vorannotierten Trainingsdaten wird anschließend nach einer Verbindung gesucht zwischen den im Bild vorhandenen Objekten bzw. dominanten Bereichen und den semantischen Konzepten, welches ein Bild als Gesamtes darzustellen versucht.

In [FGL07] wird der Ansatz weiterentwickelt und eine Konzept-Ontologie vorgestellt, die einem Teilbereich von WordNet entspricht. Hieraus können hierarchische Beziehungen zwischen gefundenen Objekten ausgelesen und die Annotation ggf. angepasst werden. Die Bilder können dank der hierarchischen Beziehungen der Konzept-Ontologie auch auf mehreren verschiedenen Ebenen annotiert werden.

In [MR05] wird ein verallgemeinertes Vorgehen zur Extrahierung von Information aus diversen multimedialen Inhalten vorgestellt. Zuerst werden Merkmale extrahiert, geclustert, und anschließend wird versucht, mittels Beziehungen zwischen Mustern unter Zuhilfenahme eines Bayes Netzwerks auf hervorstechende und auffallende Konzepte zu schließen. Diese Konzepte bzw. Zuordnungen können später zur Annotation verwendet werden.

In [AYV07] wird ein System zur automatischen Annotation von Bildern mit dem Namen Supervised Annotation by Descriptor Ensemble (SADE) vorgestellt, dessen Aufbau in Abbildung 4.22 dargestellt ist. Bilder im System werden durch eine Sammlung von Merkmalen repräsentiert. Für jedes dieser Merkmale wird ein Klassifikator anhand annotierter Beispielbilder trainiert. Jeder dieser Klassifikatoren liefert anschließend eine Ausgabe, welcher Klasse das Bild zugeordnet werden kann. Auf einer zweiten Stufe wird anschließend ein Meta-Klassifikator eingesetzt, der die verschiedenen Ausgaben der Merkmalsklassifikatoren vereint. Alle Klassifikatoren basieren auf der fuzzy kNN Methode. Ein wesentlicher Vorteil ist, dass das System problemlos durch neue Merkmale ergänzt werden kann. Außerdem können die einzelnen Merkmalsextraktoren und -Klassifikatoren auf mehrere Rechner verteilt werden. Bei der Evaluation schneidet SADE im Vergleich mit ALIP besser ab.

In [MGPW05] wird ein Klassifikationsverfahren für Bilder vorgestellt, das robust gegen Skalierungen und Betrachtungsrichtungen sein soll. Der Ansatz basiert auf zufällig gewählten quadratischen Bildbereichen (Subwindows), denen mittels annotierter Trainingsbilder Labels zugeordnet werden. Anschließend werden Extremely Randomized Trees (eine Art

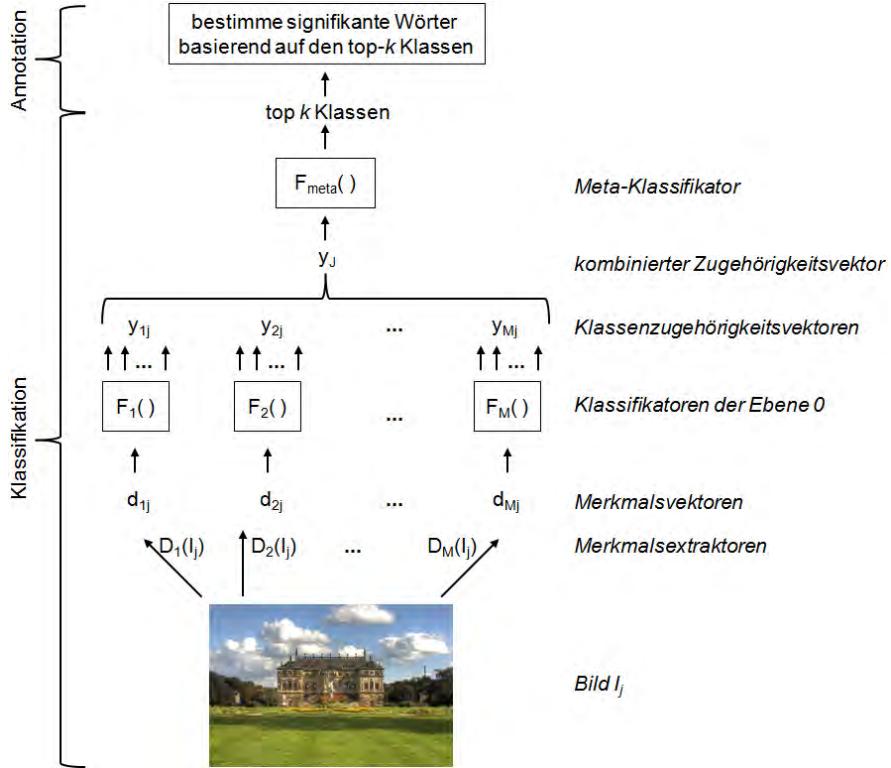


Bild 4.22: Ablauf der Bildannotation in SADE nach [AYV07].

Sammlung von Entscheidungsbäumen, genaue Beschreibung in [GEW06]) erstellt, die später für die automatische Annotation von Bildern verwendet werden können.

In [YSR05] wird ein Framework zur automatisierten Annotation von Bildern vorgestellt. Das Framework verwendet hierzu ein Modell zur Beschreibung der Verteilung bzw. Dichte von Bildmerkmalen. Als Distanzfunktion wird die Earth-Mover Distanz verwendet. Ein ähnlicher Ansatz wird in [CCMV07] verfolgt, welches ein probabilistisches Verfahren zur Annotation und Retrieval von Bildern vorstellt. Die Bilder werden als Bags von lokalen Merkmalsvektoren dargestellt. Anschließend wird eine vermischte Dichte für jedes Bild abgeschätzt. Alle diese Dichten der Trainingsbilder mit dem gleichen annotierten semantischen Konzept werden mittels einer Variante des Expectation-Maximization Algorithmus (EM) nach [DLR77] zu einer Dichteabschätzung für die korrespondierende semantische Klasse zusammengeführt.

Schließlich sollten noch einige Ansätze zur automatischen Verbesserung von Bildannotations er wähnt werden.

In [JCS04] wird eine Verbesserung für die automatische Bildannotation vorgeschlagen, indem Zusammenhänge zwischen Wörtern berücksichtigt werden, um eine genauere

Annotation zu erreichen. Das vorgestellte Coherent Language Model ist in der Lage aktiv zu lernen, d. h. es sind am Anfang weitaus weniger vorannotierte Beispielbilder nötig. In [JKWA05] werden mit verschiedenen konzeptbezogenen Ähnlichkeitsmaßen unter Einbezug von WordNet falsche bzw. fehlerhafte Annotationswörter bei Bildern eliminiert. Die Ergebnisse werden mittels der Dempster-Shafer Theorie zu einem Gesamtergebnis kombiniert. Nach eigenen Messungen verbessert sich durch das Verfahren die Genauigkeit der Annotationen erheblich.

In [WJZZ06] wird ein Graph bestehend aus Konzepten und Zahlen, die das gemeinsame Auftreten von Konzepten beschreiben, aufgebaut. Anschließend wird ein Random Walk with Restarts Algorithmus angewendet, welches als Ausgabe eine Topliste von Konzepten ausgibt. Schließlich werden die besten N Wörter als Annotation übernommen. Diese Methode wurde in [WJZZ07] erweitert und verbessert. Mittels eines auf Markov-Ketten basierenden Algorithmus werden unter Berücksichtigung des Bildinhalts die bereits existierenden Annotationen neu geordnet und die besten N Wörter beibehalten. Nach einer eigenen Auswertung – wobei die initialen Annotationen mittels der in [JLM03] vorgestellten Methode erstellt wurden – soll das Verfahren bzgl. Precision und Recall besser sein als das in [JKWA05] und [WJZZ06].

4.2.4 Andere Aufteilungen

Im Anschluss an die Aufteilung der bisherigen Ergebnisse ausgehend vom Anteil des manuellen Eingriffs werden die genannten Veröffentlichungen auch nach weiteren Kriterien eingeordnet. Eine Möglichkeit ist es die Annotationsverfahren basierend auf der Quelle des Wissens zu trennen. Demnach können manuelle Beschreibungen, Social Tagging, der umliegende Textkontext und der eigentliche Bildinhalt (Ähnlichkeit zu annotierten Trainingsbildern) unterschieden werden. Aus algorithmischer Sicht können probabilistische und nicht-probabilistische Verfahren unterteilt werden. [DJLW08] trennt die Annotationsverfahren in die gemeinsame Modellierung von Wörtern und Bildern sowie die teilweise unter manueller Beteiligung geführte Kategorisierung. Tabelle 4.9 veranschaulicht die Aufteilung der Veröffentlichungen zur Annotation von Bildern basierend auf diesen Kategorisierungen.

QUELLE	EINGRIFF	QUELLE	PROB.	NACH [DJLW08]
[AD04]	manuell	Social Tagging	–	–
[AGK ⁺ 06]	manuell	Social Tagging	–	–
[AYV07]	automatisch	inhaltsbasiert	–	Kategorisierung
[BF01]	automatisch	inhaltsbasiert	✓	Wort-Bild
[BKL ⁺ 06]	automatisch	Kontext	–	–
[CCMV07]	automatisch	inhaltsbasiert	✓	Kategorisierung
[DBFF02]	automatisch	inhaltsbasiert	✓	Wort-Bild
[DI03]	semi-autom.	inhaltsbasiert	–	Kategorisierung
[FGL07]	automatisch	inhaltsbasiert	✓	Kategorisierung
[FML04]	automatisch	inhaltsbasiert	✓	Wort-Bild
[FSN ⁺ 95]	manuell	Social Tagging	–	–
[HSWW03]	manuell	Social Tagging	–	–
[JCS04]	automatisch	(Verfeinerung)	✓	Wort-Bild
[JKWA05]	automatisch	(Verfeinerung)	✓	Wort-Bild
[JLM03]	automatisch	inhaltsbasiert	✓	Wort-Bild
[JM04]	automatisch	inhaltsbasiert	✓	Wort-Bild
[LMJ03]	automatisch	inhaltsbasiert	✓	Wort-Bild
[LW08]	semi-autom.	inhaltsbasiert	✓	Kategorisierung
[MGPW05]	automatisch	inhaltsbasiert	–	Kategorisierung
[MM04]	semi-autom.	inhaltsbasiert	–	Kategorisierung
[MR05]	automatisch	inhaltsbasiert	✓	Kategorisierung
[MTO99]	automatisch	inhaltsbasiert	✓	Wort-Bild
[PYDF04]	automatisch	inhaltsbasiert	✓	Wort-Bild
[RTMF08]	manuell	Social Tagging	–	–
[SVBM05]	automatisch	inhaltsbasiert	✓	Wort-Bild
[WDS ⁺ 01]	semi-autom.	inhaltsbasiert	–	Kategorisierung
[WJZZ07]	automatisch	(Verfeinerung)	✓	Wort-Bild
[WZJM06a]	semi-autom.	inhaltsbasiert, Kontext	✓	Wort-Bild
[YSR05]	automatisch	inhaltsbasiert	✓	Kategorisierung

Tabelle 4.9: Aufteilung der Veröffentlichungen zur Annotation von Bildern mittels verschiedener Kategorisierungen.

4.2.5 Zusammenfassung

Aus den vorangegangenen Abschnitten ist es ersichtlich, dass in den letzten Jahren viele Ideen und Ansätze zur Annotation von Bildern entstanden sind. Neben dem Ausnutzen der Eigenschaften des Social Webs versuchen die meisten Verfahren ausgehend von einer bereits annotierten Trainingsmenge inhaltsbasiert die Zuordnung von Wörtern bzw. Konzepten zu Bildern mehr oder weniger automatisch zu ermitteln. Diese Zuordnung wird anschließend zur semi- oder vollautomatischen Annotation von neuen Bildern verwendet. Bei fast allen Verfahren liegt das Hauptaugenmerk auf der Genauigkeit der Annotation, wobei die Effizienz, sowie die Erweiterbarkeit und die Skalierbarkeit vernachlässigt werden. Diese Lücken sind womöglich die Hauptgründe, warum sich die Annotationsverfahren bis heute immer noch nicht im großen Maßstab verbreiten und etablieren konnten. Für den Einsatz in einem so großen und breiten Umfeld wie das Web, aber auch für größere lokale Bildsammlungen, in denen eine Suche ermöglicht werden soll, müssen neue effiziente, erweiterbare und skalierende Lösungen erarbeitet werden.

4.3 Zusammenfassung

In Abschnitt 4.1 wurden die Grundlagen zur Objekterkennung kurz erläutert und das aktuell beste Verfahren, das BoW-Konzept, schrittweise im Detail vorgestellt. Im Wesentlichen kann festgestellt werden, dass blob- und eckenähnliche Detektoren in Kombination mit Gitternetzen und SIFT-Deskriptoren die besten Ergebnisse erzielen. Dabei existieren viele verschiedene Kombinationsmöglichkeiten für Detektoren, Deskriptoren und Grids (sog. „Kanäle“), dessen bestmögliche Auswahl und Einsatz in hierarchischen Verfahren zur Zeit noch aussteht. In vielen Ansätzen wird der Spatial-Pyramid-Ansatz aus [LSP06] in Kombination mit SVMs erfolgreich eingesetzt. Es existieren auch erste Ansätze zur effizienten und skalierenden Objekterkennung, jedoch wurde bislang die Erweiterbarkeit der Verfahren um neue Objektklassen nicht untersucht.

In Abschnitt 4.2 wurden aktuelle Ansätze zur textuellen Annotation von Bildern betrachtet. Die Verfahren sind sehr unterschiedlich, eines ist ihnen jedoch gemein: die Effizienz, die Erweiterbarkeit und die Skalierbarkeit wurden bislang kaum studiert. Für die Anreicherung der Annotation von Bildern wurde in wissenschaftlichen Veröffentlichungen bislang meistens WordNet verwendet. Der Einsatz weiterer Kategoriehierarchien sowie

die Überprüfung von deren Beitrag zur Verbesserung der Klassifikation von Objekten bzw. zur Annotation von Bildern steht zur Zeit noch aus.

Insgesamt können folgende wesentliche Probleme im Bereich der automatischen Annotation von Bildern identifiziert werden:

- die Segmentierung der Bilder in einzelne Teilbilder bzw. Objekte,
- die Genauigkeit der Erkennung von Objekten, Szenen, Personen und die daraus folgende Ableitung von Ereignissen, Aktivitäten und Orten,
- die Skalierbarkeit der Verfahren auf mehrere 100, 1 000 oder 10 000 Klassen, sowie
- die dynamische Erweiterbarkeit der Systeme um neue Klassen.

Die Segmentierung von Bildern ist selbst ein enorm breites Feld. Neben manuellen Markierungen werden Abtastungsfenster oder vollautomatische Segmentierungsalgorithmen für die Zerlegung der Bilder in Teilbilder verwendet. In dieser Arbeit wird auf die Segmentierung von Bildern nicht weiter eingegangen. Einen Überblick über verschiedene Methoden der Bildsegmentierung geben [SS01], [DJLW08], [Sze11] und [FP12].

Die nachfolgenden Kapitel stellen ein erweiterbares Verfahren für die Objekterkennung in Bildern vor, wobei eine skalierbare Lösung mit einer möglichst hohen Genauigkeit angestrebt wird.

KAPITEL 5

ERWEITERBARE OBJEKTERKENNUNGSBASIERTE ANNOTATION VON BILDERN

Die dynamische Erweiterbarkeit wurde bei Objekterkennungsansätzen bislang völlig vernachlässigt. Erweiterbarkeit ist unter dem Aspekt wichtig, dass es unwahrscheinlich ist bereits beim initialen Aufsetzen des Systems alle von [Bie87] geschätzten 30 000 Klassen zu kennen. Neue Objektklassen müssen demnach nach und nach dem System hinzugefügt werden.

Die meisten erfolgreichen Verfahren basieren auf Histogrammen von visuellen Wörtern, welche auf Basis eines statischen visuellen Vokabulars berechnet werden. Aufbauend auf diesen Histogrammen werden bei dem überwiegenden Teil der Verfahren SVMs zum Training eingesetzt, wobei jede Klasse entweder einzeln oder insgesamt mit jeder anderen Klasse verglichen wird. Sowohl das statische Vokabular als auch die Verwendung von SVMs vernachlässigen die Erweiterbarkeit der Ansätze, da beim Hinzufügen von neuen Klassen ein Neuaufsetzen des kompletten Systems nötig ist. Zwar wird das Training üblicherweise offline durchgeführt und anschließend das System auf die neue Wissensbasis umgeschaltet, bei stetig steigender Anzahl von Klassen zieht sich das Trainieren der SVMs immer mehr in die Länge.

Zwar sind Bag-of-Words-basierte Ansätze zur Zeit die erfolgreichsten auf dem Gebiet der Objekterkennung, jedoch kann deren Genauigkeit durch die Kombination mit anderen Merkmalen erhöht werden. Diese Aussage wird auch von aktuellen Ergebnissen u. a. in [VR07], [BZM08] oder [BSI08] belegt. Weiterhin werden Überlappungen und Unabhängigkeiten zwischen einzelnen Merkmalen in [Eid04] und [AFB10] untersucht. Ausgehend von diesen Kenntnissen werden in diesem Abschnitt zusätzlich weitere geeignete ergänzende Merkmale vorgestellt und auf deren Einsetzbarkeit in einem erweiterbaren System untersucht.

In diesem Kapitel werden zuerst in Abschnitt 5.1 die gängigen Klassifikationsmethoden bei Bag-of-Words-Ansätzen aus Sicht der Erweiterbarkeit untersucht. Anschließend werden in Abschnitt 5.1.1 die Probleme beim visuellen Vokabular identifiziert und erweiterbare Lösungen erarbeitet. In Abschnitt 5.1.2 wird die Anwendbarkeit von Szenebeschreibern auf Objekte sowie deren Erweiterbarkeit untersucht. Abschnitt 5.1.3 befasst sich mit der Integration von Farben bzw. Farbmerkmalen. Geeignete multidimensionale Indexstrukturen für die NN-Suche werden in Abschnitt 5.2 betrachtet und mit der sequentiellen Suche verglichen. Erkenntnisse und Ergebnisse aus diesem Kapitel werden in Abschnitt 5.3 zusammengefasst.

5.1 Klassifikation und Merkmale

Die meisten Bag-of-Words-basierten Ansätze verwenden SVMs zur Klassifikation. Eine SVM alleine ist ein binärer Klassifikator. Dementsprechend müssen bei mehr als zwei Klassen mehrere binäre SVMs trainiert werden. Bei Multiklassen-SVMs kann zwischen Abstimmung (1:1), Wettbewerb (1:N) und sukzessivem Ausschluss (graph- oder baumbasierte SVMs) unterschieden werden. Diese verschiedenen Möglichkeiten wurden in Abschnitt 4.1.7.3 näher vorgestellt.

Aus Sicht der Erweiterbarkeit ergeben sich mehrere Nachteile bei SVMs. Beim Hinzufügen einer neuen Klasse müssen je nach gewählter Methode (1:1, 1:N, DAG-SVM) so viele SVMs neu trainiert werden, wie es Klassen insgesamt im System gibt. Die benötigte Rechenzeit für das Training ist dabei enorm. In [DBLFF10] wurde ermittelt, dass bei 10 000 Klassen mit einfachem BoW allein für das Training der SVMs – unter Verwendung des effizienten LIBLINEAR-Verfahrens von [FCH⁺08] – ca. 1 CPU-Jahr nötig ist. Bei der Verwendung des Spatial Pyramid Kernels aus [LSP06] erhöht sich die Zeit für das Training der SVMs um das Hundertfache. Ein System, das nach und nach

durch neue Klassen ergänzt wird – bis die von [Bie87] geschätzten 30 000 Kategorien erreicht werden, die ein Mensch auf Anhieb erkennen kann –, ist mit den aktuellen Lösungen nicht praktikabel. Für die Klassifikation von mehreren 1 000 Klassen muss also neben einer skalierbaren auch eine erweiterbare Lösung gefunden werden.

Im Gegensatz zu SVMs haben nichtparametrische Klassifikatoren, welche ihre Klassifikationsentscheidung direkt basierend auf zugrunde liegenden Trainingsdaten bestimmen, den Vorteil, dass sie kein Lernen oder Training benötigen. Am gebräuchlichsten ist dabei die Familie der NN-Klassifikatoren [BGRS99, DHS00]. Diese basieren auf der Berechnung oder Abschätzung von Distanzen der zu klassifizierenden Objekte zu der vorhandenen Trainingsmenge. Das Objekt wird anschließend derjenigen Klasse zugeordnet, zu dem der nächste (NN) oder die k nächsten Nachbarobjekte (kNN) gehören.

Wesentliche Vorteile der nichtparametrischen Klassifikatoren im Vergleich zu lernbasierten Verfahren sind nach [BSI08]:

- Sie können natürlich mit einer großen Zahl an Klassen umgehen,
- Overfitting der Parameter wird vermieden, was bei lernbasierten Verfahren ein zentrales Problem darstellt und
- es ist keine Lern- bzw. Trainingsphase nötig.

Die Untersuchungen in [BSI08] haben gezeigt, dass NN-basierte Verfahren durchaus ähnlich gut abschneiden wie die am meisten verbreiteten SVM-Lösungen. Auch durch die Evaluation in [DBLFF10] wird bestätigt, dass NN-Ansätze bei der Verwendung einer großen Klassenmenge besser abschneiden als SVM-basierte Lösungen. Wegen dieser Vorteile sind NN-Verfahren wesentlich besser für ein dynamisch wachsendes erweiterbares System geeignet. Aus diesen Gründen wird in den nachfolgenden Abschnitten nach NN-basierten Lösungen mit wenig oder keinem Training gesucht.

5.1.1 Bag of Words und das visuelle Vokabular

Aus Sicht der Erweiterbarkeit ist das visuelle Vokabular das Hauptproblem vom BoW-Ansatz. Das allgemeine Vorgehen zur Erstellung des visuellen Vokabulars wurde in Abschnitt 4.1.2.3 beschrieben. Zwei wesentliche Probleme des visuellen Vokabulars werden in den nachfolgenden Abschnitten behandelt: Zum Einen die Bestimmung der optimalen Anzahl der visuellen Wörter im Vokabular in Abschnitt 5.1.1.1. Zum Anderen die

Probleme mit dem Einsatz eines statischen Vokabulars aus der Sicht der Erweiterbarkeit des Systems in Abschnitt 5.1.1.2.

5.1.1.1 Anzahl der visuellen Wörter im Vokabular

Die meisten BoW-basierten Ansätze verwenden den k -means Clustering Algorithmus zur Erstellung des visuellen Vokabulars. Ziel bei der Erstellung des Vokabulars ist die Reduzierung der Anzahl verschiedener Deskriptoren und dadurch das Zusammenfassen von verrauschten Varianten von ähnlichen Deskriptoren. Dazu werden die (SIFT-)Deskriptoren der Trainingsbilder (oder eine repräsentative Teilmenge davon) in k Cluster eingeordnet, wobei das Clusterzentrum die Menge der zugehörigen Merkmale durch einen Deskriptor, das sog. visuelle Wort repräsentiert.

Dabei ist die Wahl von k kritisch [JDS08] und vom Datensatz abhängig [YJHN07]. Je größer das k gewählt wird, desto kleinere Cluster ergeben sich. Dadurch erhöht sich zwar die Präzision in den Clustern, d. h. in ein Cluster kommen wirklich nur noch die verrauschten Varianten des selben Deskriptors. Leider erhöht sich dabei auch das Quantisierungsrauschen, da bei kleineren Clustern einige verrauschte Varianten des selben Deskriptors in benachbarten Clustern landen. Bei der Wahl von kleineren Werten für k vergrößern sich die Cluster. Dadurch erhöht sich zwar die Wahrscheinlichkeit, dass alle verrauschten Varianten eines Deskriptors in ein und das selbe Cluster fallen, jedoch kommen auch viele andere Deskriptoren in das selbe Cluster, was die Präzision verringert. Der Unterschied zwischen kleinen und großen k -s wird in Abbildung 5.1 an einem Beispiel dargestellt. Die Wahl des richtigen k ist somit immer ein Kompromiss zwischen dem Quantisierungsrauschen und der Präzision des Deskriptors bzw. des entstehenden visuellen Wortes. Aus diesem Grund und da zwischen dem k und der Größe des Datensatzes nach [YJHN07] ein Zusammenhang besteht, ist die dynamische Anpassung von k bei der Verwendung eines globalen Vokabulars für ein erweiterbares System essenziell.

5.1.1.2 Erweiterbarkeit des Vokabulars

Die meisten BoW-basierten Ansätze verwenden ein statisches globales visuelles Vokabular. Der wesentliche Nachteil dabei ist, dass das Vokabular nur ein einziges Mal beim Training basierend auf allen oder einem Teil der Trainingsbilder ermittelt wird. Anschließend wird dieses statische Vokabular bei allen Bildern zur Berechnung von Histogrammen von visuellen Wörtern verwendet (siehe Abschnitt 4.1.2.4). Das heißt, das visuelle Vokabular

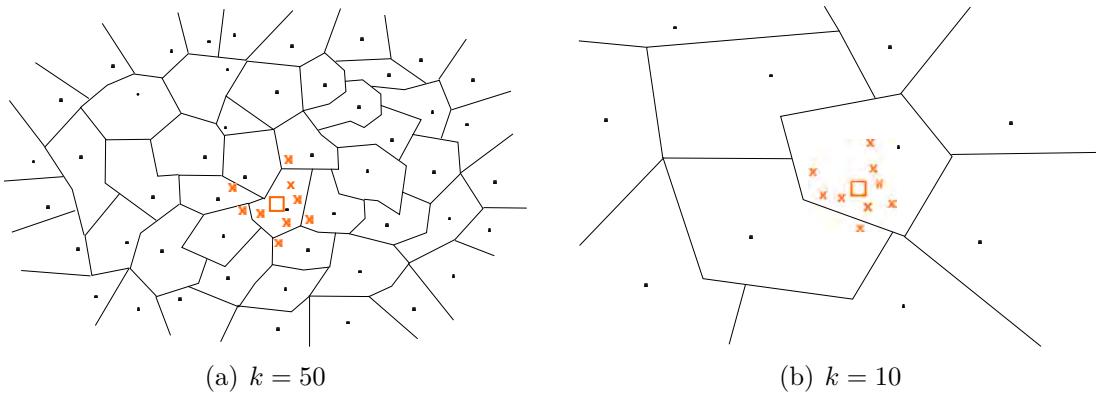


Bild 5.1: Deskriptor- und Quantisierungsrauschen in Abhängigkeit von k nach [JDS08]. Das Symbol \bullet markiert die Zentroiden der jeweiligen Cluster, \square stellt den aktuell betrachteten Deskriptor dar, \times stehen für verrauschte Versionen des Deskriptors.

wird sowohl für die Trainingsbilder als auch für die zukünftig zu klassifizierenden Bilder eingesetzt. Sofern das System um eine oder mehrere Klassen erweitert werden sollte, muss das bereits festgelegte Vokabular verwendet werden. Dabei kann die neue Klasse auch völlig abweichende visuelle Wörter beinhalten, die im statischen Vokabular nicht vorkommen bzw. nicht durch eine oder mehrere Cluster im Vokabular vertreten sind.

Übertragen auf Texte ergibt sich folgendes Problem der Erweiterbarkeit. Es liegen zwei Texte mit identischem Inhalt in zwei verschiedenen Sprachen vor. Aus beiden Texten kann jeweils ein Vokabular der Sprachen erstellt werden sowie mittels dem BoW-Ansatz zugrunde liegenden Text-Mining-Verfahren ein Wörterbuch ermittelt werden. Die Vokabulare der zwei Sprachen beinhalten – verständlicherweise – nur Begriffe, die in den Texten vorkommen. Das Wörterbuch kann durch die Hinzunahme von neuen Textfragmenten (identische Texte verfasst in den zwei Sprachen) erweitert werden. Da die neuen Textfragmente auch neue Begriffe enthalten können, die in den Vokabularen bislang nicht auftauchen und auch nicht durch Begriffe in den Vokabularen beschrieben werden können, müssen die Vokabulare erweitert werden. Aus diesem Grund muss auch im visuellen Fall beim Hinzufügen einer neuen Klasse ggf. das globale visuelle Vokabular erweitert werden. Ab einer gewissen Anzahl von Klassen kann es dabei dazu kommen, dass alles visuell erfassbare bereits im visuellen Vokabular vorhanden ist und keine Erweiterung mehr nötig ist bzw. erfolgen kann. Wann jedoch dieser Fall eintritt und woran festgestellt werden kann, ob alles erfasst wurde und wie groß dieses allgemeine visuelle Vokabular sein wird, ist unbekannt (vgl. Abschnitt 5.1.1.3).

Als Beispielszenario wurde ein visuelles Vokabular basierend auf Bildern von Früchten (Apfel, Banane und Traube) mit $k = 300$ erstellt. In einem nächsten Schritt soll das System durch die neue Klasse „Motorrad“ erweitert werden. Beispielhafte visuelle Wörter aus dem Vokabular der Früchte sowie interessante Punkte aus einem Motorradbild sind in Abbildung 5.2 dargestellt.

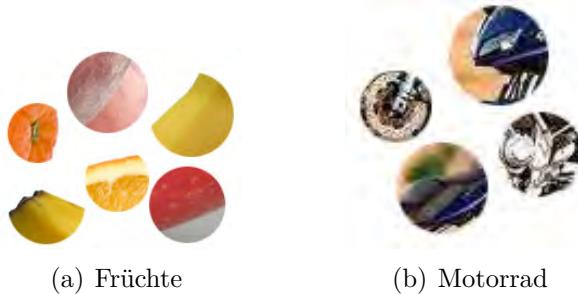


Bild 5.2: Beispielhafte visuelle Wörter aus dem visuellen Vokabular für Früchte und interessante Punkte eines Motorradbildes.

Schon bei diesen Ausschnitten ist der Unterschied zwischen visuellen Wörtern des Vokabulars und den Deskriptoren der neuen Klasse offensichtlich. Noch deutlicher wird der Unterschied beim Erstellen der Histogramme der visuellen Wörter. Abbildung 5.3(a) zeigt das Histogramm für das Foto einer Weintraube, Abbildung 5.3(b) für das Bild eines Motorrads. Die Zuordnung der Deskriptoren von Motorräder zu fast jedem visuellen Wort des Früchte-Vokabulars zeigt, dass die Deskriptoren von Motorräder schlecht den visuellen Wörtern des Vokabulars zuordenbar sind.

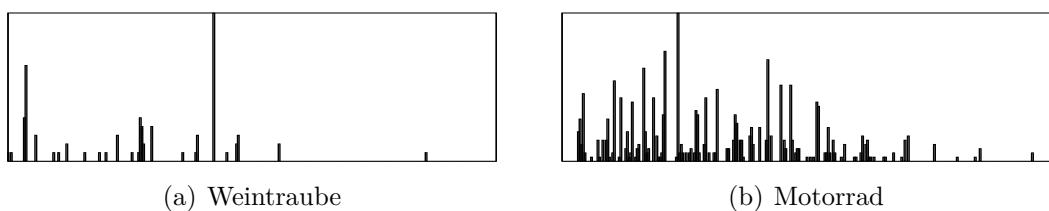


Bild 5.3: Histogramme von visuellen Wörtern für ein Foto einer Weintraube und eines Motorrads unter Verwendung des Früchte-Vokabulars.

Offenbar sind die visuellen Wörter des Vokabulars nicht diskriminativ genug für neue Klassen, somit müsste das visuelle Vokabular angepasst werden. Möglichkeiten zur Anpassung des visuellen Vokabulars in einem erweiterbaren System werden im folgenden Abschnitt vorgestellt.

5.1.1.3 Möglichkeiten für erweiterbare Vokabulare

Unter Berücksichtigung der zwei in Abschnitt 5.1.1.1 und Abschnitt 5.1.1.2 vorgestellten wesentlichen Probleme bei BoW-basierten Ansätzen bzgl. des visuellen Vokabulars werden in diesem Abschnitt Möglichkeiten für erweiterbare Vokabulare untersucht [NMW10]. Folgende Lösungen können in Betracht gezogen werden:

1. Verwendung eines statischen allgemeinen visuellen Vokabulars,
2. Anpassung des visuellen Vokabulars bei
 - Hinzunahme einer neuen Klasse oder
 - nachdem die Diskriminativität des Vokabulars unter eine festgelegte Grenze fällt,
3. Einsatz klassenspezifischer Vokabulare statt eines allgemeinen Vokabulars sowie
4. Vokabular vollständig weglassen und direkt auf den Deskriptoren die NN-Suche durchführen.

Der wesentliche Vorteil bei der Verwendung eines statischen allgemeinen visuellen Vokabulars ist, dass sich das Vokabular bei Hinzunahme einer neuen Klasse nicht ändert. Allerdings bleiben bei dieser Lösung mehrere Fragen unbeantwortet: Wann ist ein visuelles Vokabular „allgemein“ genug? Wie soll ein allgemeines Vokabular erstellt werden? Wie soll die Auswahl an Bildern aussehen, aus denen das allgemeine Vokabular erstellt wird? Wie groß soll das allgemeine Vokabular sein, damit später ca. 30 000 Objekte (wie in [Bie87] geschätzt) klassifiziert werden können? In [YJHN07] wurde gezeigt, dass die optimale Größe des Vokabulars bzgl. der Genauigkeit der Klassifikation je nach Datensatz anders ausfällt. Die aufgeführten Fragen können ohne Kenntnis der zukünftigen Klassen nicht beantwortet werden.

Abhilfe könnte die zweite Lösungsidee schaffen, bei der das allgemeine Vokabular bei Hinzunahme einer neuen Klasse oder beim Erreichen eines Schwellenwertes angepasst wird. Wesentliche Vorteile von diesem Ansatz sind, dass das visuelle Vokabular Wörter für alle Klassen enthält und bei der Anpassung des Vokabulars ggf. auch die Anzahl der visuellen Wörter vergrößert. Allerdings ist die Anpassung des visuellen Vokabulars mit hohen Kosten verbunden. Da beim BoW-Ansatz die Trainingsbilder durch Histogramme von visuellen Wörtern repräsentiert werden und diese wiederum auf dem visuellen Vokabular aufbauen, müssen beim Hinzufügen einer neuen Klasse für alle Bilder jeder Klasse die Histogramme von visuellen Wörtern neu berechnet werden. Daraus folgt

einerseits, dass das Training der Klassifikatoren komplett neu beginnen muss, da diese auf den Histogrammen aufbauen. Andererseits ist auch die Abspeicherung von allen Deskriptoren für alle Bilder nötig, da diese zur Berechnung der Histogramme erforderlich sind.

Eine weitere Lösungsidee aus Sicht der Erweiterbarkeit ist statt einem globalen statischen Vokabular je Klasse ein eigenes visuelles Vokabular zu verwenden. Die Klassifikation kann anschließend direkt auf diesen klassenspezifischen Vokabularen aufsetzen und ggf. auch Häufigkeiten der visuellen Wörter mit einbeziehen. Klassenspezifische Vokabulare wurden in [FSMST05, PDCB06, PD07, Per08] eingesetzt, allerdings nur in Kombination mit statischen globalen Vokabularen. Die Erweiterbarkeit der Ansätze wurde zwar als mögliches Potenzial erwähnt, jedoch nicht weiter detailliert ausgeführt – womöglich deswegen, da genau das globale Vokabular die Erweiterbarkeit verhindert.

Die Verwendung von klassenspezifischen Vokabularen unabhängig von einem globalen Vokabular hat die Vorteile, dass die Berechnung der Histogramme von visuellen Wörtern entfällt. Es ist kein allgemeingültiges Vokabular vorhanden und dadurch sind auch keine Anpassungen des Vokabulars nötig. Weiterhin besteht so die Möglichkeit unterschiedliche Vokabulargrößen für die einzelnen Klassen zu verwenden, was der Forderung aus Abschnitt 5.1.1.1 nachkommt. Zur Beschreibung von Kometen z. B. reichen weniger visuelle Wörter aus als zum Beschreiben eines Fahrrads oder Autos. Der Nachteil bei diesem Ansatz ist, dass es kaum erforscht ist und somit erst Experimente nötig sind, um die Eigenschaften und optimale Parameter zu bestimmen.

Als letzte Alternative ergibt sich die Möglichkeit Vokabulare komplett wegzulassen und die Klassifikation direkt auf den (SIFT-)Deskriptoren aufzusetzen. Dieser Ansatz wird z. B. in [BSI08] und [AF10] verfolgt, wobei beide den nächsten Nachbardeskriptor aus der Trainingsmenge für das zu klassifizierende Bild suchen. Nach den Messungen von [BSI08] schneidet dieses Verfahren im Vergleich mit den auf statischen globalen visuellen Vokabularen und SVM-Klassifikatoren aufbauenden Ansätzen ähnlich gut und z. T. sogar besser ab. Wesentlicher Vorteil der genannten Verfahren ist, dass weder ein visuelles Vokabular noch die darauf aufbauenden Histogramme von visuellen Wörtern berechnet werden müssen. Es ist auch keine Lern- oder Trainingsphase nötig, da direkt auf den Merkmalen mittels einer NN-Suche gearbeitet wird. Damit sind diese Lösungen sehr leicht erweiterbar. Allerdings sind die Kosten für die Suche nach dem nächsten Nachbardeskriptor sehr hoch, da jeder Deskriptor des zu klassifizierenden Bildes mit jedem Deskriptor der Trainingsmenge verglichen und die Distanz dazu berechnet werden muss.

Bei einem erhöhten Datenvolumen, d. h. bei mehreren 1 000 Klassen, muss von erheblichen Berechnungszeiten für die Klassifikation ausgegangen werden. Abhilfe können dabei entweder Reduktionslösungen bzw. Filter (wie in [AF10]) oder geeignete Indexstrukturen (siehe Abschnitt 5.2) schaffen.

Im folgenden Abschnitt wird der Lösungsweg unter Einsatz eines klassenspezifischen Vokabulars verfolgt, da dieser Ansatz sowohl akzeptable Berechnungszeiten bei der Klassifikation als auch einen geringen Speicherbedarf sowie gute Erweiterbarkeitsmöglichkeiten verspricht.

5.1.1.4 Klassenspezifische visuelle Vokabulare

Ausgehend von der Bewertung der erweiterbaren Möglichkeiten für visuelle Vokabulare in Abschnitt 5.1.1.3 wird in diesem Abschnitt eine Lösung basierend auf klassenspezifischen visuellen Vokabularen erarbeitet.

Aus einer top-down Perspektive betrachtet kann ein globales Vokabular für verschiedene Klassen aufgespalten werden, wobei dann für jedes zu klassifizierende Bild je visuelles Vokabular ein Histogramm von visuellen Wörtern erstellt werden müsste. Dieser Ansatz kann auch als Abwandlung der Methode von [PD07] gesehen werden, wo bipartite Histogramme eingesetzt werden, um festzustellen, ob ein Bild zu einer Klasse gehört oder eher zu anderen Klassen zugeordnet werden soll. Beim aktuellen Szenario würden sich statt bipartite n-partite Histogramme ergeben. Die Idee der Zerlegung kann quasi bis zu einer beliebigen Tiefe rekursiv weitergeführt werden, wobei dann noch speziellere visuelle Vokabulare entstehen. Als Beispiel könnte man sich die weitere Aufspaltung des Vokabulars für Früchte in Äpfel, Bananen, Trauben, Orangen, etc. vorstellen. Abbildung 5.4 veranschaulicht ein Objekthierarchie für die Aufspaltung von visuellen Vokabularen.

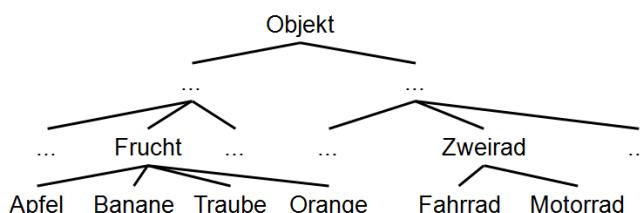


Bild 5.4: Beispiel Objekthierarchie zur top-down bzw. bottom-up Perspektive für den Zusammenhang zwischen dem globalen und den klassenspezifischen Vokabularen.

Aus der bottom-up Perspektive betrachtet könnten zuerst die visuellen Vokabulare für jede Klasse einzeln getrennt berechnet werden. Durch die Vermischung der einzelnen

klassenspezifischen Vokabulare können gruppierte Vokabulare erstellt werden. Wird die Vermischung rekursiv fortgesetzt, gelangt man im letzten Schritt zu einem allgemeinen globalen Vokabular, ähnlich wie in [FSMST05] oder [PDCB06]. Letztendlich müssten für ein Bild jedoch weiterhin so viele Histogramme von visuellen Wörtern berechnet werden, wie viele Vokabulare im Zweig der Hierarchie existieren. Ein weiteres Problem ist, dass beim Hinzufügen einer neuen Klasse ggf. alle Vokabulare vom neuen Blatt bis zur Wurzel angepasst werden müssten und durch die Änderung der Wurzel – also des allgemeinen Vokabulars – für jedes Bild im System neue Histogramme von visuellen Wörtern berechnet werden müssten. Das würde im Anschluss auch das erneute Training der Klassifikatoren mit sich ziehen.

Dem geschilderten Gedankengang folgend wurde eine neue Lösung basierend auf klassenspezifischen Vokabularen entwickelt, welche im folgenden Abschnitt im Detail vorgestellt wird.

Unabhängige klassenspezifische visuelle Vokabulare

Dem geschilderten Gedankengang folgend entstand die Idee vollkommen unabhängige klassenspezifische Vokabulare für jede Klasse separat zu berechnen. Wegen der hohen Variabilität der Darstellung ein und des selben Objekts und der unterschiedlichen Hintergründe in den Bildern wird angenommen, dass für die Beschreibung der Klasse nur ein Teil der gefundenen interessanten Punkte in den Bildern tatsächlich relevant sind. Deswegen werden nur diejenigen visuellen Wörter im klassenspezifischen Vokabular behalten, welche am häufigsten in den Trainingsbildern der Klasse auftauchen. Visuelle Wörter, welche nicht zu den häufigsten Wörtern einer Klasse gehören, werden als visuelle Outlier-Wörter bezeichnet. Die Hypothese ist, dass später bei der Klassifikation von Testbildern Bilder, die zu einer gegebenen Klasse gehören, weniger visuelle Outlier-Wörter besitzen werden als Bilder, die zu anderen Klassen gehören.

In gewisser Hinsicht vereinigt dieser Ansatz die Histogramme von visuellen Wörtern mit visuellen Vokabularen. Die häufigsten visuellen Wörter einer Klasse werden mit ihrer durchschnittlichen Häufigkeit und dem Radius des Clusters als Klassenbeschreibung abgespeichert. Die so entstandenen klassenspezifischen Vokabulare können auch als eine Repräsentation des Durchschnittsbildes einer Klasse angesehen werden. Zur Bestimmung der zugehörigen Klasse für ein zu klassifizierendes Bild wird eine spezielle Scoring-Funktion vorgeschlagen, welche auf Ideen aus klassischen Bildretrieval-Verfahren unter Berücksichtigung der visuellen Outlier-Wörter konzipiert wurde.

Berechnung der Klassenbeschreibungen

In diesem Abschnitt wird beschrieben, wie die im vorherigen Abschnitt konzipierten klassenspezifischen Vokabulare erstellt werden.

Zuerst werden die Bilder auf eine einheitliche Größe skaliert, da bei den meisten Detektoren die Anzahl der gefundenen interessanten Punkte von der Bildgröße abhängig ist (siehe Abschnitt 4.1.2.1). Sofern ein Trainingsbild wesentlich größer ist als alle anderen Bilder, kann dies die Berechnung des visuellen Vokabulars und die durchschnittliche Häufigkeit der visuellen Wörter wesentlich beeinflussen. Das feste Seitenverhältnis sollte keine Auswirkungen auf die detektierten interessanten Punkte haben, sofern der eingesetzte Detektor und der Deskriptor skalierungsunabhängig sind. Die Bilder wurden nach [TMF04] und [Tor09] auf 128x128 Pixel skaliert.

Die Bilder werden anschließend in den HSV-Farbraum konvertiert und dessen V-Kanal, welcher die Helligkeitsinformationen enthält, extrahiert. Im nächsten Schritt werden mittels eines festen Gitternetzes oder unter Verwendung von Detektoren interessante Punkte im Bild ermittelt und durch SIFT-Deskriptoren beschrieben. Für die Ermittlung der interessanten Punkte und der Berechnung der SIFT-Deskriptoren wurden in dieser Arbeit die Merkmalsextraktoren von [MS05, MTS⁺05]¹ als Teil des erweiterbaren Merkmalsextraktionswerkzeugs verwendet, welches in Abschnitt 7.2.1 näher vorgestellt wird. Unter Verwendung des Hessian-Affine-Detektors ergeben sich für den Caltech 256 Datensatz von [GHP07] mit auf 128x128 Pixel skalierten Bildern im Durchschnitt 275 SIFT-Deskriptoren pro Bild.

Die ermittelten SIFT-Deskriptoren der Trainingsbilder einer Klasse werden anschließend mittels des k -means Clustering Algorithmus unter Verwendung der L2-Distanz in k Cluster eingeordnet. Statt der L2-Distanz können auch beliebige andere Distanzen eingesetzt werden. Danach werden die Häufigkeiten für jedes visuelle Wort in der Trainingsmenge berechnet und die visuellen Wörter des Vokabulars absteigend sortiert. Die am häufigsten auftretenden n visuellen Wörter ($n < k$) werden behalten, die restlichen Wörter verworfen. Die bestmöglichen Werte für n und k bzw. deren optimales Verhältnis zueinander wird in Abschnitt 6.1 bestimmt. Anschließend werden die Mittelpunkte jedes Clusters und deren Radius (größte Distanz zwischen dem Mittelpunkt und den Elementen des Clusters) berechnet. Für jede Klasse werden die häufigsten n visuellen Wörter, die

¹ <http://www.robots.ox.ac.uk/~vgg/research/affine/>

durchschnittliche Anzahl derer Vorkommen in einem Bild der Klasse und der Radius des Clusters als Klassenbeschreibung abgespeichert. Abbildung 5.5 veranschaulicht das vorgestellte Verfahren zur Berechnung der Klassenbeschreibungen.

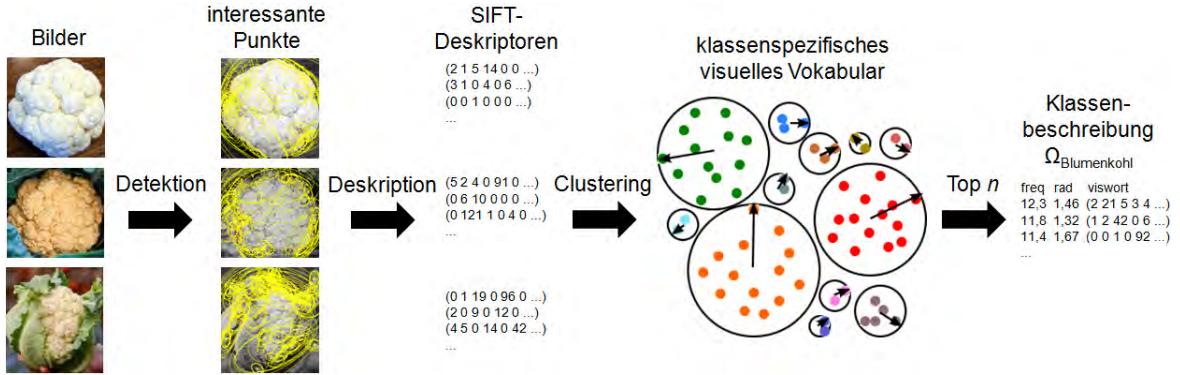


Bild 5.5: Berechnung des klassenspezifischen Vokabulars und der Klassenbeschreibung für die Klasse $\Omega_{\text{Blumenkohl}}$ anhand einer Stichprobe.

Im Wesentlichen kann diese Klassenbeschreibung als die Zusammenführung des visuellen Vokabulars mit dem Histogramm von visuellen Wörtern aufgefasst werden. Sie bietet dabei eine Repräsentation eines Durchschnittsbildes der gegebenen Klasse. Die Klassenbeschreibung kann auch als Fingerabdruck oder ID der Klasse bzw. des jeweiligen Objekts angesehen werden.

Scoring-Funktion und Klassifikation

Die ersten Schritte bei der Berechnung des Score-Wertes für ein zu klassifizierendes Bild sind identisch mit denen bei der Erstellung der Klassenbeschreibungen. Das Bild muss zuerst auf die selbe einheitliche Größe skaliert werden, die auch bei den Trainingsbildern eingesetzt wurde. Anschließend wird das Bild in den HSV-Farbraum konvertiert, der V-Kanal extrahiert, interessante Punkte detektiert und diese durch SIFT-Deskriptoren beschrieben. Zur Förderung der Lesbarkeit werden nachfolgend die Bilder, die Mustern $f(\mathbf{x})$ entsprechen, durch I abgekürzt.

Um die Ähnlichkeit zwischen einem Bild I und einer Klasse Ω_κ (bzw. Klassenbeschreibung) zu bestimmen, wurde eine neue Scoring-Funktion konzipiert. Die Idee dabei ist jeden Deskriptor eines Bildes I zu den n visuellen Wörtern der Beschreibung der Klasse Ω_κ zuzuordnen. Bei der Bestimmung des am nächsten gelegenen visuellen Wortes sollte das gleiche Distanzmaß verwendet werden wie auch beim Clustering. Sofern ein Deskriptor des Bildes I am nächsten zu einem visuellen Wort von Ω_κ liegt, die Distanz jedoch den maximalen Radius des Clusters überschreitet, wird der gegebene Deskriptor

als visuelles Outlier-Wort behandelt und einer speziellen Outlier-Gruppe $n+1$ zugeordnet. Dabei wird angenommen, dass die Anzahl der Outlier-Deskriptoren für Bilder der selben Klasse niedriger sein wird als für Bilder, die zu einer anderen Klasse gehören. Da auch der Bildhintergrund Outlier-Deskriptoren hervorbringen kann, ist es nicht ausreichend allein ausgehend von der Existenz von Outlier-Deskriptoren ein Bild bzgl. einer Klasse auszuschließen. Es ist viel mehr erstrebenswert die Outlier-Deskriptoren mit einer Bestrafung (penalty) in die Scoring-Funktion zu integrieren.

Nachdem alle Deskriptoren des Bildes I den visuellen Wörtern von Ω_κ bzw. der Outlier-Gruppe zugeordnet wurden, werden für alle Gruppen die Anzahl der Deskriptoren ($freq(I_i)$) und die durchschnittliche Distanz in der Gruppe ($avgdist(I_i, \Omega_{\kappa_i})$) berechnet. Aus der Beschreibung der Klasse Ω_κ kann die durchschnittliche Häufigkeit eines visuellen Wortes i abgelesen werden, sie wird als $freq(\Omega_{\kappa_i})$ bezeichnet. Für die spezielle Outlier-Gruppe wird dieser Wert $freq(\Omega_{\kappa_{n+1}})$ auf 0 gesetzt. Die durchschnittliche Distanz der Outlier-Gruppe ergibt sich aus dem Durchschnitt der Distanzen der Deskriptoren ($avgdist(I_{n+1}, \Omega_{\kappa_{n+1}})$). Diese durchschnittliche Distanz ist meistens ein hoher Wert, da die Outlier-Deskriptoren wegen der zu großen Distanz keinem visuellen Wort zugeordnet werden konnten. Zusätzlich wird die durchschnittliche Distanz der Outlier-Gruppe mit der Anzahl der Outlier-Deskriptoren ($freq(I_{n+1})$) gewichtet. Somit ist bereits eine Bestrafung für Outlier-Deskriptoren mit in die Scoring-Funktion integriert.

Basierend auf den erläuterten Termen wird der Score für ein Bild I und eine Klasse Ω_κ durch die Funktion in Gleichung 5.1 ermittelt. Abbildung 5.6 veranschaulicht die Berechnung des Score-Werts für ein gegebenes Bild I und eine Klasse Ω_κ .

$$score(I, \Omega_\kappa) = \frac{1}{n+1} \sum_{i=1}^{n+1} |freq(\Omega_{\kappa_i}) - freq(I_i)| \cdot avgdist(I_i, \Omega_{\kappa_i}) \quad (5.1)$$

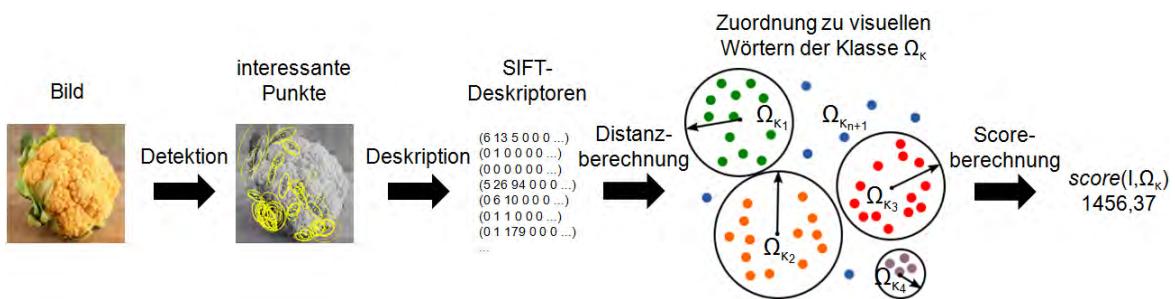


Bild 5.6: Berechnung des Score-Werts für ein gegebenes Bild I und der Klasse Ω_κ ($score(I, \Omega_\kappa)$). Die blauen Punkte repräsentieren visuelle Outlier-Wörter und sind der speziellen Outlier-Gruppe $\Omega_{\kappa_{n+1}}$ zugeordnet.

Der Score-Wert wird für das Bild I für jede Klasse $\Omega_\kappa \in \Omega$ im System berechnet. Je kleiner der Wert $score(I, \Omega_\kappa)$, desto ähnlicher ist ein Bild I zur Klasse Ω_κ . Das Bild wird der Klasse Ω_κ mit dem kleinsten Score-Wert zugewiesen (siehe Gleichung 5.2).

$$\Omega^* = \arg \min_{\Omega_\kappa} score(I, \Omega_\kappa) \quad (5.2)$$

Im vorgestellten Ansatz sind mehrere Parameter näher zu untersuchen, wie die optimale Wahl von k , das beste Verhältnis zwischen n und k sowie deren Abhängigkeit von der Anzahl der verwendeten Trainingsbilder. Die Bestimmung der Parameter wird in Kapitel 6 durchgeführt.

5.1.1.5 Zusammenfassung

Zwei wesentliche Probleme des visuellen Vokabulars wurden in diesem Abschnitt detailliert vorgestellt sowie mögliche Lösungswege diskutiert. Anschließend wurde die beste Option aus Sicht eines erweiterbaren Systems weiter verfolgt und ein Ansatz basierend auf klassenspezifischen Vokabularen vorgestellt. Der Ansatz ist durch neue Klassen einfach erweiterbar und ermöglicht insbesondere auch die parallele Berechnung der Score-Werte. Als Ergänzung zum BoW-basierten Ansatz werden im folgenden Kapitel weitere Merkmale auf ihren Einsatz in einem erweiterbaren System untersucht.

5.1.2 Objekte als Szenen

Als Alternative bzw. Ergänzung zum Verfahren aus Abschnitt 5.1.1 wird in diesem Abschnitt der mögliche Einsatz von Szenendeskriptoren für die Erkennung von Objekten im Rahmen eines erweiterbaren Systems untersucht.

[Hen05] definiert den Begriff einer Szene als eine semantisch zusammenhängende, benennbare, auf den Menschen zugeschnittene Sicht einer Umgebung in der realen Welt. Diese Sicht besteht meistens aus Hinter- und Vordergrundelementen, die in einer hierarchischen räumlichen Ordnung angeordnet sind. Basierend auf dieser Definition ist die Trennung zwischen Objekten und Szenen abhängig von der räumlichen Skalierung. Als Beispiel kann man sich einen Tisch vorstellen, der Teil einer Büro-Szene sein kann. Wenn jedoch näher an den Tisch herangegangen wird, verwandelt sich die Tischplatte selber in eine Szene mit Elementen, wie z. B. Stifte, Papier, Umschläge oder ein Telefon.

Ausgehend von dieser Definition können Objekte nach deren Blickwinkeln geordnet ebenfalls als Szenen interpretiert werden. Im Folgenden werden die zusätzlichen blickwinkelabhängigen Aufteilungen von Klassen $\Omega_\kappa \in \Omega$ als Subklassen $\Omega_{\kappa_i} \in \Omega_\kappa$ bezeichnet. Für die Klasse Fahrrad können die Subklassen vorne, rechts, links und hinten, oder noch feiner abgestuft zusätzlich in vorne-rechts, vorne-links, hinten-rechts, hinten-links definiert werden. Abbildung 5.7 zeigt eine Beispielhierarchie basierend auf dem PASCAL VOC 2007 [EVG⁺07] Datensatz, ergänzt durch Subklassen für die Klasse Fahrrad.

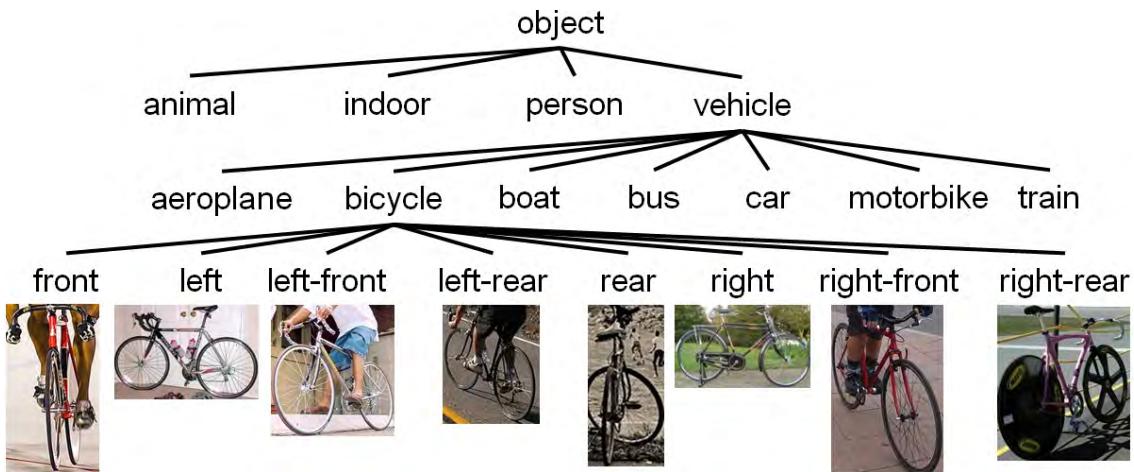


Bild 5.7: Blickwinkelbasierte Subklassen für die Klasse Fahrrad eingebettet in die Objekt-hierarchie des PASCAL VOC 2007 Datensatzes.

Die meisten Bilder in diesen Subklassen weisen untereinander eine beachtliche Ähnlichkeit auf. Sie besitzen den gleichen räumlichen Aufbau, bestehen aus den gleichen Vordergrundelementen und der Hintergrund ist auch häufig ähnlich oder auf einige unterschiedliche Hintergründe beschränkt. Objekte können in Anlehnung an die Szenen-definition von [Hen05] damit auf dieser Betrachtungsebene als Szenen aufgefasst werden. Abbildung 5.8 zeigt zur Verdeutlichung einige Beispiele der Subklasse Fahrrad_{vorne}.



Bild 5.8: Beispielbilder der Subklasse Fahrrad_{vorne}. Zwischen den Bildern ist eine starke Ähnlichkeit bzgl. räumlichem Aufbau sowie Vorder- und Hintergrundelementen feststellbar, somit können Objekte auf dieser Betrachtungsebene auch als Szenen aufgefasst werden.

5.1.2.1 Szenendeskriptor für Objekte

Basierend auf diesem Gedankengang wird die Verwendbarkeit des erfolgreichen GIST-Szenendeskriptors für Objekte untersucht. Der GIST-Deskriptor wurde allgemein in Abschnitt 4.1.2.2 kurz vorgestellt. Im Wesentlichen entspricht GIST einem SIFT-Deskriptor, der auf das ganze Bild angewendet wird. Mehrere Untersuchungen [MTEF06, RTL⁺07, Tor03, WTF08, KG09] haben für GIST sehr gute Erkennungsraten bei Szenen nachgewiesen. In [DJS⁺09, HE08b, HE08a, TFW08] wurde auch die Eignung von GIST für den Einsatz in sehr großen Bilddatenbeständen gezeigt, vor allem für das Auffinden von ähnlichen Szenen und Duplikaten. Aus diesem Grund scheint GIST ein geeignetes Merkmal für ein System mit sehr vielen Klassen zu sein.

5.1.2.2 Berechnung der Klassenbeschreibungen

Für die Repräsentation einer Subklasse wird eine Klassenbeschreibung basierend auf den GIST-Deskriptoren der Trainingsbilder berechnet. Eine Klasse wird dabei durch die Gesamtheit der Klassenbeschreibungen ihrer Subklassen repräsentiert. Von den Trainingsbildern wurde nur der annotierte Bereich betrachtet, welcher das gegebene Objekt genau umrahmt. Die Annotationen dazu stammten in dieser Arbeit aus dem PASCAL VOC 2007 Datensatz. Die GIST-Deskriptoren wurden aus auf 32x32 Pixeln skalierten Bildern mittels der Implementierung von [DJS⁺09] berechnet. Zur Reduzierung der Größe der Klassenbeschreibungen und der Anzahl der Vergleiche bei der nachfolgenden Klassifikation wurden kompaktere Darstellungen evaluiert: Zum Einen wurde der Mittelwert aller GIST-Deskriptoren der Subklassen berechnet (meanGIST). Zum Anderen wurde der am nächsten zum Mittelwert liegende GIST-Deskriptor als Repräsentant seiner Subklasse vermerkt (medianGIST). Da alle Subklassenbeschreibungen unabhängig von einander sind, ist dieses Verfahren für ein erweiterbares System gut geeignet.

5.1.2.3 Ablauf der Klassifikation

Zur Klassifikation eines neuen Bildes wird zuerst das Bild durch manuelle Markierungen, Abtastungsfenster oder durch aktuelle vollautomatische Segmentierungsalgorithmen in einzelne Teilbilder zerlegt. In dieser Arbeit wurden die manuellen Annotationen des PASCAL VOC 2007 Datensatzes und in einem separaten Test das Segmentierungsverfahren von [AMFM09], welches dem aktuellen Stand der Technik entspricht, eingesetzt.

Für jedes Teilbild werden GIST-Deskriptoren mit der Implementierung von [DJS⁺09] ermittelt. Anschließend werden die Distanzen zu den Mittelwerten, Medianen bzw. allen GIST-Deskriptoren der Subklassen mittels der L1- und L2-Distanz berechnet. Das Teilbild wird der Subklasse und der zugehörigen übergeordneten Klasse mit der niedrigsten Distanz zugewiesen. Das bedeutet, dass mit der am nächsten gelegenen Subklasse Motorrad_{vorne} auch die übergeordnete Klasse Motorrad zum Teilbild zugewiesen wird. Die vollständige Annotation des Bildes ergibt sich im Anschluss durch die Vereinigungsmenge der Klassenzuordnungen der einzelnen Teilbilder.

5.1.2.4 Zusammenfassung

In diesem Abschnitt wurde ein Verfahren zum Einsatz des erfolgreichen Szenendeskriptors GIST zur Objekterkennung vorgestellt. Der Ansatz basiert auf der Szenendefinition von [Hen05] und fasst blickwinkelabhängige Darstellungen von Objekten als Szenen auf. Das geschilderte Verfahren wird in Abschnitt 6.2 ausführlich evaluiert und optimiert.

5.1.3 Farben

Sowohl die vorgestellten SIFT- als auch die GIST-Deskriptoren werden auf Graustufenbildern angewendet. Obwohl beim menschlichen Sehempfinden verschiedene Abstufungen von Grautönen bereits sehr viel Information in sich tragen, kann auch die Farbe von Objekten und Szenen die Erkennung beeinflussen. Das menschliche Sehorgan reagiert dabei vor allem auf die Gegensätze von unmischbaren Farben wie z. B. grün und rot oder blau und gelb. Diese Erkenntnisse bilden die Grundlage u. a. von diversen Bildkomprimierungsverfahren sowie vom $L^*a^*b^*$ -Farbraum (L steht für Luminanz, a und b für Chrominanz), bei dem sich die Distanzen zwischen den Farbkoordinaten an den physiologischen Eigenschaften der menschlichen Farbwahrnehmung orientieren [Hof09]. In diesem Abschnitt werden die Einsatzmöglichkeiten und Auswirkungen von Farbmerkmalen untersucht.

5.1.3.1 Farb-SIFT Deskriptoren

Alle Farbvarianten des SIFT-Deskriptors basieren auf der Idee je Farbkanal einen SIFT-Deskriptor zu extrahieren und diese anschließend zu einem einzigen Vektor zu konkatenieren. Je nach gewähltem Farbraum ergeben sich unterschiedliche Farb-SIFT-Deskriptoren.

Die unterschiedlichen Farbvarianten von SIFT-Deskriptoren wurden in [SGS10] mit weiteren farbbasierten Merkmalen, wie z. B. Farbmomenten und Farbhistogrammen auf deren Eignung für die Objekterkennung evaluiert. Dabei übertrumpften die SIFT-Deskriptoren die anderen Merkmale deutlich. Auch die Hinzunahme von Farbinformation hat die Erkennung um bis zu 8% im Vergleich zu traditionellen Graustufen-basierten SIFT-Deskriptoren verbessert. Allerdings wurden für diese Verbesserung alle Farbvarianten gleichzeitig eingesetzt, was die Berechnungszeit erheblich verlängert. Im direkten Vergleich der Farb-SIFT-Deskriptoren hat sich herausgestellt, dass je nach verwendetem Datensatz die Reihenfolge der Deskriptoren bzgl. der Genauigkeit der Objekterkennung unterschiedlich ausfällt.

Aus Sicht der Erweiterbarkeit bieten die Farb-SIFT-Deskriptoren die gleichen Vor- und Nachteile, wie sie in Abschnitt 5.1.1 diskutiert wurden.

5.1.3.2 Bag of Colors

Statt visuellen Wörtern, also SIFT-Deskriptoren von interessanten Punkten, verwendet [WDJ11] einen „Sack voller Farben“ (Bag of Colors (BoC)). Im Prinzip handelt es sich dabei um ein Farbhistogramm mit mehreren Verbesserungen im Vergleich zur klassischen Version. Als Farbraum wurde $L^*a^*b^*$ verwendet, welches bei der Evaluation durchweg bessere Resultate lieferte als Histogramme im RGB-Farbraum. Für das Histogramm wurde keine gleichmäßige Aufteilung des Farbraums verwendet, sondern zuerst aus 10 000 zufällig ausgewählten Bildern von Flickr ein Farbvokabular erstellt. Der Ablauf dazu ist ähnlich wie auch beim Erstellen des visuellen Vokabulars bei BoW. Im Anschluss wird jedes Pixel eines Bildes dem nächstliegendem Farbwert aus dem Farbvokabular zugeordnet und somit ein Histogramm von Farben erstellt.

Bei der Evaluation in [WDJ11] wurde gezeigt, dass die Verwendung eines erlernten Farbvokabulars wesentlich bessere Ergebnisse liefert als die gleichmäßige Aufteilung des Farbraums. Als weitere Verbesserung im Vergleich zu gewöhnlichen Farbhistogrammen wurde der sog. Power-Law aus [PSM10] angewendet. Dabei werden alle Werte eines Vektors $\mathbf{x} = (x_1, \dots, x_d)$ durch deren Quadratwurzel ersetzt ($x_i := \sqrt{x_i}$), wodurch das Farbhistogramm geglättet wird. Zuletzt wurde noch eine L1-Normalisierung durchgeführt, um Histogramme unter sich vergleichbar zu machen, d. h. die Werte des Vektors wurden durch $x_j := \frac{x_j}{\sum_{i=1}^d x_i}$ ersetzt.

Aus Sicht der Erweiterbarkeit lässt sich das BoC-Verfahren sehr gut in Objekterkennungssystemen einsetzen. Zwar ist das Erlernen eines Farbvokabulars nötig, um wesentlich bessere Erkennungsraten zu erreichen als mit traditionellen Farbhistogrammen, ein einmalig aus Fotos erlerntes Farbvokabular scheint jedoch universell genug zu sein für allgemeine Fotos. Im Vergleich zu visuellen Vokabularen (vgl. Abschnitt 5.1.1) kann diese Eigenschaft auch dadurch begründet werden, dass der Merkmalsraum bei Farben wesentlich kleiner ist. Farben werden durch 3 Werte beschrieben, während SIFT-Deskriptoren meistens durch einen Vektor mit 128 Dimensionen repräsentiert werden.

Weiterhin ist das BoC-Merkmal sehr schnell und leicht zu berechnen. Der Deskriptor für ein Bild ist wesentlich kleiner als bei BoW, wo entweder jeder SIFT-Deskriptor abgespeichert wird oder die Histogramme von visuellen Wörtern bei größeren Datensätzen gerne mehrere tausend Bins besitzen.

Eine Evaluation des BoC-Merkmales erfolgt in Abschnitt 6.3.

5.1.4 Kombination von Merkmalen

Viele Objekterkennungsansätze und die meisten Bildretrieval-Verfahren verwenden eine Kombination von mehreren Merkmalen, welche jeweils unterschiedliche Eigenschaften der Bilder erfassen. Bezogen auf die gewünschte Klassifikation und abhängig von der Zusammensetzung der Klassenmenge (d. h. wie ähnlich bzw. wie unterschiedlich die Bilder einer Klasse unter sich bzw. zwischen Klassen sind) leisten einige Merkmale einen größeren Beitrag zur Klassifikation als andere. Unterschiedliche und komplementäre Merkmale müssen somit durch ein geeignetes Verfahren basierend auf der Menge der Trainingsbilder möglichst optimal bzgl. der Genauigkeit der Klassifikation kombiniert werden.

In diesem Abschnitt werden Möglichkeiten zur Kombination von Merkmalen kurz erläutert und im Anschluss ein Verfahren ausgewählt, welches bei der Optimierung des vorgestellten erweiterbaren Ansatzes in Abschnitt 6.4 eingesetzt wird.

Eine Möglichkeit ist die einfache Konkatenation der einzelnen Merkmalsvektoren. Anschließend kann auf diesem Vektor entweder direkt oder nach der Reduktion der Dimensionalität ein Klassifikator trainiert werden [YYZL03]. Die Erweiterbarkeit durch neue Klassen hängt bei dieser Kombinationsmöglichkeit von der eingesetzten Klassifikationsmethode ab. Eine inkrementelle Ergänzung durch neue Merkmale ist jedoch nur durch den kompletten Neuaufbau des Systems möglich.

Als weitere Kombinationsmöglichkeit können Klassifikatoren zuerst auf einzelnen Merkmalen oder Merkmalsgruppen trainiert und anschließend die Ergebnisse der einzelnen Klassifikatoren mittels Mehrheitsabstimmung oder durch ein hierarchisches Verfahren in ein einziges Endergebnis verschmolzen werden. [KHD98] vergleicht verschiedene Möglichkeiten zur Kombination von Klassifikatoren. Ein hierarchisches Verfahren wird in [JJ93] vorgestellt. Diese Möglichkeit der Kombination bietet große Freiheiten bzgl. der Gewichtung der einzelnen Klassifikatoren und der zugrundeliegenden Merkmale. Außerdem erlaubt es auch die Kaskadierung von Klassifikatoren, welche bei einer geschickten Anordnung die benötigte Zeit für die Klassifikation verkürzen können. Ein weiterer Vorteil ist dabei auch die relativ leichte Integration neuer Merkmale in das System zu einem späteren Zeitpunkt.

Viele Objekterkennungsverfahren verwenden SVMs für die Klassifikation. Multiklassen-SVMs sind lediglich eine Kombination von binären SVMs, welche z. B. im 1:N-Fall (Wettbewerb) eine Klasse von allen anderen Klassen abtrennen. Im Bereich der Objekterkennung hat sich für die Kombination von unterschiedlichen Merkmalen in SVMs in den letzten Jahren das MKL-Verfahren nach [LCB⁺04, BLJ04] verbreitet. Im Wesentlichen kann MKL als eine lineare Kombination von verschiedenen Kernfunktionen aufgefasst werden, welche als Gesamtheit die Klassifikation verbessern. Trotz der Optimierung des Lernprozesses in [SRSS06, GN09] und erfolgreicher Anwendung von MKL in der Objekterkennung z. B. in [KS07] oder [VR07] bleibt ein wesentlicher Nachteil bzgl. der Erweiterbarkeit. Da jede Klasse mit allen anderen Klassen verglichen wird, müssen bei der Erweiterung des Systems die SVMs für die komplette Klassenmenge neu erlernt werden. Bei einer steigenden Anzahl von Klassen wächst der Lernaufwand ebenfalls mit und ist nach [DBLFF10] bei mehreren 1 000 Klassen nicht tragbar.

5.1.4.1 Optimierungsmethoden

In dieser Arbeit werden die optimalen Gewichte w_i für die Merkmale $\mathbf{c} \in \{\text{cs-BoW Dense, cs-BoW HesAff, GIST, Bag of Colors}\}$ gesucht, für welche die MAP bei einem Datensatz mit mehreren 1 000 Objektklassen maximiert wird.

Da die Wertelandschaft der Gewichte recht breit und der Pfad zum Ergebnis nicht relevant ist, können für die Bestimmung der Gewichte Optimierungsalgorithmen der lokalen Suche herangezogen werden. [NR03] nennt folgende wesentliche Optimierungsalgorithmen der lokalen Suche:

- Bergsteigeralgorithmus (hill-climbing),
- simulierte Abkühlung (simulated annealing),
- lokale Strahlensuche (local beam search),
- genetische Algorithmen.

Der Bergsteigeralgorithmus beginnt die Suche von einem zufällig gewählten Startpunkt der Wertelandschaft, berechnet die Funktion für die direkten Nachbarpunkte und wählt die beste Richtung anhand der Zielfunktion (in der Analogie zum Bergsteigen die höchste Steigung). Der Algorithmus terminiert, sofern keiner der Nachbarn eine Verbesserung bietet. Die ursprüngliche Version des Algorithmus kann jedoch bei lokalen Maxima, Plateaus oder Bergrücken stecken bleiben und das globale Maximum verfehlten. Eine Abhilfe diesbezüglich schafft die Variante mit zufällig gewählten Neustarts (auch random-restart oder shotgun hill-climbing genannt). Nach einer vorher festgelegten Anzahl von Schritten oder beim Steckenbleiben des Bergsteigers wird eine neue Iteration mit einem zufällig ausgewählten Startpunkt begonnen. In [GSK98] wurde nachgewiesen, dass die zufällige Bestimmung der Startpunkte die Lösungsfindung begünstigt. Das Verfahren wurde vielfältig erweitert und verfeinert. Bei der von [GL97] entwickelten „Tabu“-Suche wird eine Liste von k bereits untersuchten Zuständen mitgeführt, welche nicht erneut besucht werden dürfen. Der STAGE-Algorithmus von [BM98] approximiert die Wertelandschaft anhand der ersten Durchläufe des random-restart hill-climbing Algorithmus und setzt die Suche beim approximierten globalen Maximum fort.

Die simulierte Abkühlung wurde von [KGV83] in Anlehnung an [MRR⁺53] konzipiert. Die Idee des Optimierungsverfahrens verfolgt die Abkühlung von Metallen, dessen Atome mit kontrolliert sinkender Temperatur sich in einem möglichst energiearmen Zustand verhärteten. Der Ablauf ist dem Bergsteigeralgorithmus ähnlich, jedoch wird nicht der beste, sondern ein zum Teil zufälliger nächster Schritt ausgewählt. Dadurch sind auch Bewegungen vom Optimum hinweg möglich. Die Wahrscheinlichkeit der Auswahl von solchen Bewegungen nimmt mit der sinkenden Temperatur ab, ähnlich wie bei der Abkühlung von Metallen die Freiheitsgrade von Atomen eingeschränkt werden.

Die lokale Strahlensuche [NR03] beginnt mit k zufälligen Startpunkten und berechnet jeden Nachbarn für alle k Zustände. Aus dieser Menge werden die k besten ausgewählt und der Vorgang wiederholt. Ein Nachteil dieses Verfahrens ist, dass sich die k Zustände schnell innerhalb einer Region der Wertelandschaft sammeln, wodurch das Verfahren zu einer etwas teureren Variante des Bergsteigeralgorithmus mutiert. Dieser Nachteil kann

umgangen werden, indem nicht die besten k nächsten Zustände weiter verfolgt, sondern mit einer Wahrscheinlichkeit p auch weniger gute Folgezustände beigemischt werden.

Genetische Algorithmen [SP94] basieren auf der Lehre von Charles Darwin. Durch die natürliche Selektion überleben von einer Population nur die stärksten Mitglieder. Die Stärke von einzelnen Individuen der aktuellen Population wird durch eine sogenannte Fitness-Funktion berechnet. Basierend auf den so ermittelten Werten werden die stärksten Mitglieder ausgewählt (Selektion) und durch Rekombination der Zustände neue Individuen erzeugt. Durch zufällige Mutationen können neue Eigenschaften in die Population einfließen und der Vorgang kann von Neuem beginnen.

In dieser Arbeit wird in Abschnitt 6.4 der random-restart hill-climbing Algorithmus für die Ermittlung der optimalen Gewichtung und Kombination von Merkmalen verwendet, da dieser Algorithmus in [GMS08] bereits erfolgreich für die Objekterkennung eingesetzt wurde.

5.2 Effiziente Suche

Damit die Klassifikation und dadurch die Annotation von Bildern möglichst schnell erfolgt, ist eine effiziente merkmalsbasierte Suche nötig. Zur Realisierung können multidimensionale Indexstrukturen eingesetzt werden, wozu [Sam06] und [BBK01] einen umfassenden Überblick geben. Eine weitere Möglichkeit zur Steigerung der Effizienz besteht in der Kompression der Merkmalsvektoren oder im Einsatz von approximativen Suchverfahren, bei denen jedoch auch immer ein Verlust in der Genauigkeit einhergeht. Das wesentliche Ziel dabei ist durch die Minimierung der Zugriffe und Vergleiche die NN-Suche schneller zu gestalten und dadurch das Erkennungssystem zumindest bzgl. der Antwortzeit auf mehrere 1 000 Klassen skalierbar zu machen. Des Weiteren gilt zu beachten, inwieweit die Verfahren für ein erweiterbares System geeignet sind.

5.2.1 Indizierung

Für hochdimensionale Merkmalsräume können nach [BBK01, WSB98] im Wesentlichen Datenpartitionierende (Data Partitioning (DP)) und Raumpartitionierende (Space Partitioning (SP)) Indexstrukturen unterschieden werden, von denen in Multimedia Datenbanken meistens DP-basierte verwendet werden [Kos04, Kos10]. SP-Indexstrukturen teilen den Merkmalsraum mittels vorher festgelegten Gitterlinien auf, unabhängig von

eventuellen Clustern in der Datenmenge. Beispiele für SP-Indizes sind kd-Bäume, Quad-Bäume oder Gridfiles. DP-basierte Methoden unterteilen den Merkmalsraum hierarchisch um die Kosten der Suche von $O(n)$ auf $O(\log n)$ zu verringern. Für die einzelnen unterteilten Regionen können Hyperwürfel und Hypersphären unterschieden werden. Zu DP-basierten Indexstrukturen zählen unter anderem R-Bäume (sowie seine Varianten), X-, TV-, M-, SS- und SR-Bäume.

Im Rahmen des GiST-Projekts von [HNP95] wurde eine einheitliche Schnittstelle für den Einsatz und den Vergleich verschiedener multidimensionaler Indexstrukturen erstellt. In den nachfolgenden Jahren wurden mehrere Indexstrukturen in das GiST-Framework integriert, so dass aktuell B-, R-, R*-, SS- und SR-Bäume unterstützt werden. Alle Bäume können dynamisch durch neue Einträge erweitert werden und kommen somit für ein erweiterbares System in Frage. In dieser Arbeit wurde das in [Wöl11] angepasste und bereinigte GiST Framework, die kd-Baum Implementierung von [Mou10] sowie die original SR-Baum-Implementierung¹ von [KS97] eingesetzt.

In [KS97] wurde der SR-Baum eindeutig als bestes Verfahren zur Indizierung von multidimensionalen Daten für die NN-Suche unter den genannten Indexstrukturen herausgestellt. Vergleichsgrößen waren dabei die Anzahl der E/A-Operationen (i. Allg. Plattenzugriffe) und die benötigte CPU-Zeit. Zwar sind die genannten Indexstrukturen für multidimensionale Merkmale gedacht, leider sinkt deren Effizienz jedoch rapide mit steigender Zahl der Dimensionen. Nach [Sam06] sind SS- und SR-Bäume in gleichverteilten hochdimensionalen Räumen ([Sam06] nennt 32 oder 64 Dimensionen) langsamer als ein sequentieller Scan, da jeder Blattknoten im Baum gelesen werden muss. In [WSB98] wurde gezeigt, dass je nach eingesetzter Indexstruktur die Grenze für das Lesen jedes Blattknotens bereits bei 30 bis 50 Dimensionen erreicht wird.

Um diesen Nachteil von multidimensionalen Indexstrukturen zu umgehen, wurden verschiedene Lösungen vorgeschlagen. Im Wesentlichen basieren die Ansätze auf der Approximierung des nächsten Nachbarn, der Komprimierung der Indexstrukturen sowie der Aufteilung der Dimensionen.

¹ <http://www dbl nii ac jp/~katayama/homepage/research/srtree/English.html>

5.2.1.1 Baum-Striping

In [BBK⁺00] wurde eine an die Idee von RAID-Systemen angelehnte Aufteilung („Striping“) von Indizes auf mehrere getrennte Bäume vorgeschlagen. Dabei wird nicht der Merkmalsraum selber partitioniert (vgl. horizontale Fragmentierung in verteilten Datenbanksystemen [Rah96]), sondern es werden durch die Aufteilung der d Dimensionen auf k disjunkte Mengen k Subräume mit niedrigerer Dimensionalität erzeugt (vgl. vertikale Fragmentierung in verteilten Datenbanksystemen). Um die Zuordnung der so entstehenden sog. Subobjekte eindeutig gewährleisten zu können, wird zusätzlich für jedes Subobjekt der Objektidentifikator abgespeichert. Die disjunkt vertikal zerlegten Merkmale werden durch mehrere – potenziell auch unterschiedliche – multidimensionale Indexstrukturen indiziert. Eine Anfrage wird ebenfalls vertikal zerlegt, auf den einzelnen Indizes ausgewertet und die Teilergebnisse zusammengeführt. Ein Beispiel für die Zerlegung einer Anfrage ist in Abbildung 5.9 abgebildet.

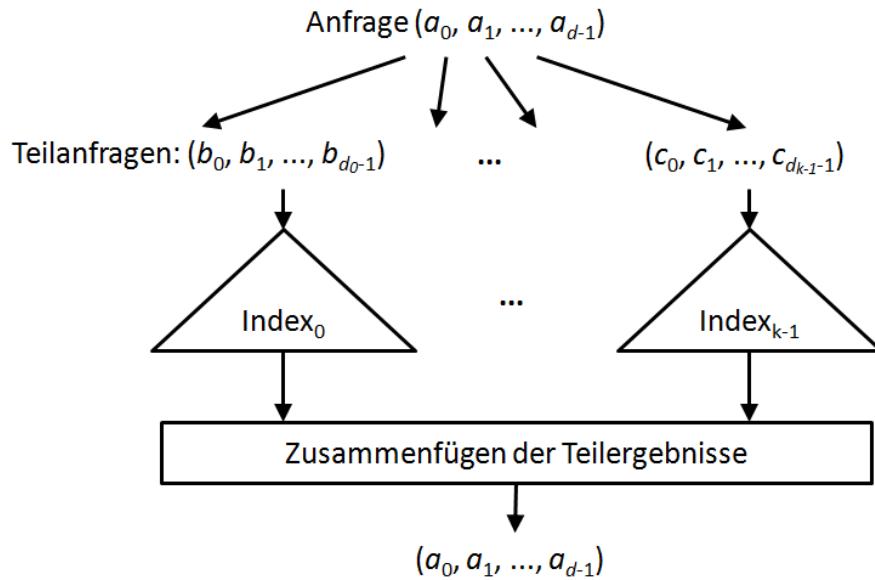


Bild 5.9: Vertikale Zerlegung einer Anfrage mittels Tree Striping nach [BBK⁺00].

Ausgehend von der in [BBK⁺00] vorgestellten Kostenfunktion, welche sowohl die verteilte Anfrage und das Zusammenfügen der Teilergebnisse berücksichtigt, wurde gezeigt, dass ein optimales k zur Aufteilung der Dimensionen in Abhängigkeit von der Anzahl N der indizierten Vektoren, der Anzahl der Dimensionen d und der Anzahl C_{eff} der in einer Seite des verwendeten Index abgespeicherten Vektoren berechnet werden kann. Nach [BBK⁺00] ist die vertikale Zerlegung erheblich effizienter als die Verwendung eines d -dimensionalen Index oder d eindimensionaler invertierten Listen. Das Verfahren

lässt sich auch sehr gut auf mehrere Rechner verteilen, da die Anfragen auf den einzelnen Subindizes unabhängig voneinander bearbeitet werden können.

Leider wurde das vorgestellte Verfahren nur für Bereichsanfragen und nicht für die NN-Suche evaluiert. Der wesentliche Nachteil des vorgestellten Verfahrens aus Sicht der Erweiterbarkeit ist, dass das optimale k abhängig von der Anzahl der Vektoren N ist und sich somit in einem erweiterbaren System stetig ändert. Das hat zur Folge, dass die vertikale Partitionierung neu erstellt wird und dadurch die Indizes komplett neu aufgebaut werden müssen.

5.2.2 Approximative Nächste-Nachbarn-Suche

Sofern eine Annäherung an den nächsten Nachbarn für die Anwendung ausreicht, können approximative nächste Nachbarn (ANN) Verfahren eingesetzt werden, wobei die Verkürzung der Suchzeit mit der Verschlechterung der Qualität der Ergebnisse einhergeht. ANN-Verfahren können nach [Sam06] und [BAG03] wie folgt unterteilt werden:

1. Unterbrechung der NN-Suche nach dem Lesen von einer vordefinierten Anzahl von Blöcken oder Zellen,
2. Einschränkung der Suche durch den zusätzlichen Parameter ε , so dass $d(q,o') \leq (1 + \varepsilon) \cdot d(q,o)$ erfüllt ist, wobei q das Anfrageobjekt und o' das innerhalb von $(1 + \varepsilon)$ der Distanz zu dem wirklichen nächsten Nachbarn o liegende Objekt ist.

Nachfolgend werden die wichtigsten ANN Verfahren kurz vorgestellt.

5.2.2.1 Cluster Pruning

Bereits in [SW78] wurde ein Ansatz basierend auf hierarchischem Clustering für die Indizierung von hochdimensionalen Daten vorgeschlagen. Wirklich aufgegriffen wurde es erst durch die Festlegung der theoretischen Grundlagen und Messungen in [CPR⁺07] bzgl. dem Einsatz des Cluster-Pruning-Verfahrens für die NN-Suche. Die Methode ist denkbar einfach: Ausgehend von den zu indizierenden n Merkmalsvektoren werden zufällig \sqrt{n} sog. Anführer („Leader“) ausgewählt und die restlichen Datenpunkte zu ihren nächstgelegenen Clusterzentren zugeordnet. Dadurch erhält man eine Partitionierung des Merkmalsraumes, der je nach Verteilung der Datenpunkte meistens recht gleichmäßig ausfällt. In [CPR⁺07] wurde nachgewiesen, dass bei einer konstanten Anzahl von Dimensionen d , n Datenpunkten und \sqrt{n} Clustern die Anzahl der Elemente in den

Clustern mit hoher Wahrscheinlichkeit höchstens $\sqrt{n} \log n$ sein wird. Bei der ANN-Suche werden die Distanzen zunächst nur zu den Anführern berechnet und anschließend in der Partition (bzw. Cluster) des nächstgelegenen Anführers weitergesucht. Das Verfahren kann rekursiv fortgesetzt werden, was einem hierarchischen Clustering ähnlich wie in [NS06] nahe kommt. Optional können neben dem nächstgelegenen Cluster auch noch weitere nächste Nachbarcluster hinzugenommen sowie die Zuordnung der Datenpunkte auf mehrere statt nur einem Cluster erweitert werden. Somit lässt sich auch die Qualität der approximierten nächsten Nachbarn einstellen.

Das Cluster-Pruning-Verfahren aus [CPR⁺07] wurde in [GJA10] für die ANN-Suche in großen Bilddatensätzen eingesetzt und erweitert. Die Bilder wurden durch einen SIFT-ähnlichen Deskriptor beschrieben, welche anschließend mittels Cluster Pruning partitioniert wurden. Zusätzlich wurde die Festlegung der initialen Anzahl der Cluster abgeändert, um die entstehenden Clustergrößen genau auf die gewünschte Blockgröße des Speichers zu optimieren. Als weiterer Parameter wurde die Möglichkeit eingeführt zusätzliche Cluster hinzuzunehmen. Diese werden mit den kleineren Clustern zusammen erneut geclustert, um die Clustergrößen möglichst gleichmäßig zu halten. Im Wesentlichen vergrößert sich dadurch die Zeit für die Berechnung der Cluster, die Zeit für die ANN-Suche verringert sich jedoch bei gleich guter Qualität der Suchergebnisse. Bei der Evaluation wurden zu allen Deskriptoren von insgesamt 3 120 Anfragebildern die $k = 20$ nächsten Nachbarn gesucht und das am häufigsten vorkommende Bild als Antwort zurückgegeben. Die Ergebnisse auf Datensätzen mit 20 bzw. 189 Millionen Deskriptoren (30 000 bzw. 300 000 Fotos) zeigen beide eine Genauigkeit von knapp 75%. Die Zeit für das Clustering nimmt mit wachsender Größe erheblich zu (36-fach), während die Zeit zur Bearbeitung einer Anfrage gut skaliert (2,2-fach).

Aus Sicht der Erweiterbarkeit wurde der Cluster-Pruning-Ansatz bislang noch nicht untersucht. In einem stetig wachsendem System tritt ein ähnliches Problem wie auch bei visuellen Vokabularen auf, da die Cluster dynamisch wachsen müssen. Als Lösungsstrategien können ähnliche genannt werden wie die in Abschnitt 5.1.1.3 vorgestellten. Ein regelmäßiger erneuter Aufbau der Cluster ist nicht praktikabel, da mit der Vergrößerung des Datensatzes die Zeit für die Berechnung der Cluster erheblich zunimmt.

5.2.2.2 Merkmalsbasierte Vorselektion

In dieser Arbeit wird eine Idee in Anlehnung an Pruning-Verfahren basierend auf einer Teilmenge der verwendeten Merkmale eingesetzt. In der Regel beeinflusst bei der

Kombination der Merkmale eine Teilmenge die Klassifikation und damit das Endergebnis stärker. Sofern die Vergleiche mittels dieser Merkmale sogar schnell berechnet werden können, ist diese Teilmenge von Merkmalen für eine Vorselektion der relevanten Klassen bestens geeignet.

Der Ansatz beruht auf der Annahme, dass bei einer hinreichend großen Anzahl von Klassen im System für ein gegebenes Testbild irrelevante Klassen bereits durch eine Teilmenge der Merkmale ausgeschlossen werden können, ohne dabei die Qualität der Annotation stark zu verringern. Somit kann eine Vorselektion der potenziellen Klassen ermittelt und im weiteren Schritt mittels teurerer Vergleiche nur noch auf einem eingeschränkten Suchraum nach nächsten Nachbarn gesucht werden. In dieser Funktionsweise ähnelt der Ansatz dem Filtern von potenziellen Bereichen bei der fensterbasierten Segmentierung von Bildern wie z. B. in [VJ01]. Insgesamt verschnellert sich dadurch die Klassifikation, wobei – ähnlich zu den bislang vorgestellten approximativen Verfahren – ebenfalls zwischen dem Maß der eingeführten Ungenauigkeit und dem Performance-Gewinn abgewogen werden muss. Bei Bedarf können auch mehrere Zwischenstufen zur Einschränkung des Suchraums eingeführt werden.

Für die Auswahl bzw. Einstufung der Merkmale bzgl. deren Eignung zur schnellen und mit möglichst wenig Verlusten versehenen Einschränkung des Suchraums sind zwei wesentliche Aspekte zu berücksichtigen: Einerseits muss das Merkmal alleine eine möglichst hohe Genauigkeit bei der Klassifikation aufweisen. Diese Eigenschaft kann entweder durch die Berechnung der MAP bzgl. einer gegebenen Menge von Klassen oder durch die prozentuale Überlappung der nächsten k Nachbarn im Vergleich mit der kombinierten Anwendung von allen Merkmalen festgestellt werden. Andererseits muss das gegebene Merkmal auch schnell mit den Klassenbeschreibungen vergleichbar sein, was durch einfache Laufzeitmessungen ermittelt werden kann. Entsprechende Messungen und Optimierungen diesbezüglich werden in Abschnitt 6.5 durchgeführt.

5.2.3 Komprimierung der Merkmalsvektoren

Eine weitere Möglichkeit für eine schnellere NN-Suche ist die Komprimierung der Merkmalsvektoren. Dabei werden die Merkmalsvektoren in einen neuen Merkmalsraum mit weniger Dimensionen abgebildet. Einerseits können dazu gängige Dimensionsreduktionsverfahren wie z. B. PCA oder die diskrete Kosinustransformation (Discrete Cosine Transform (DCT)) eingesetzt werden. Andererseits können sog. embedding Methoden

verwendet werden [Sam06], welche die Anzahl der Vergleiche für die NN-Suche reduzieren. Hierbei wird eine Ähnlichkeitsmatrix für alle in der Trainingsmenge enthaltenen Merkmalsvektoren erstellt. Anschließend werden die Merkmale in einem neuen Vektorraum eingebettet, so dass die Distanzen so gut wie möglich erhalten bleiben. Meistens kann dieser Schritt als eine Quantisierung der Merkmalsvektoren angesehen werden, wie z. B. in der Darstellung des VA-File in [WSB98].

Eine spezielle Art von embedding Methoden gekoppelt mit der ANN-Suche ist das in letzter Zeit sehr populär gewordene Locality Sensitive Hashing (LSH) [GIM99], [AI08]. Dabei werden die Merkmalsvektoren zuerst in einen Hamming-Würfel abgebildet. Anschließend werden mehrere randomisierte Hashfunktionen auf den Vektoren ausgewertet, was im Prinzip der mehrfachen Quantisierung der Vektoren entspricht. Das Ziel dabei ist nahe gelegene Vektoren in die gleichen Buckets abzubilden und dadurch die Distanzen zwischen den einzelnen Vektoren beizubehalten. In [PJA10] wurden unterschiedliche Hashfunktions-Klassen speziell aus der Sicht der Bildsuche unter Einsatz des SIFT-Deskriptors betrachtet. Hashfunktionen, welche auf unstrukturierten Quantisierern basieren, schnitten dabei besser ab, da diese auch die Verteilung der Daten berücksichtigen. Aus den untersuchten Hashfunktionen hat sich ein k -means-basierter Ansatz als bestes Verfahren herausgestellt.

Aus Sicht der Erweiterbarkeit ist der wesentliche Nachteil bei allen Komprimierungsstrategien, dass die Trainingsmenge vollständig bekannt sein muss, um die Distanzen zwischen den Trainingsvektoren berechnen und komprimieren zu können. Bei einem stetig wachsenden System muss somit die Komprimierung bei jeder Erweiterung erneut vollständig durchgeführt werden, was bei einer immer größer werdenden Anzahl von Klassen immer mehr Zeit benötigt und somit nicht praktikabel ist.

5.2.4 Zusammenfassung

In diesem Abschnitt wurden Verfahren für die effiziente NN-Suche vorgestellt. Die in Abschnitt 5.2.1 vorgestellten multidimensionalen Indexstrukturen bieten leider nur bis zu einigen 10 Dimensionen einen Vorteil. Bei Merkmalen wie z. B. SIFT oder GIST mit 128 bzw. 960 Dimensionen ist die sequentielle Suche schneller, so dass auf den Einsatz von multidimensionalen Indexstrukturen verzichtet wird. Kompressionsverfahren für Merkmalsvektoren aus Abschnitt 5.2.3 sind direkt von der verwendeten Stichprobe abhängig und somit für ein dynamisch erweiterbares System ungeeignet. Bei Methoden

zur ANN-Suche aus Abschnitt 5.2.2 muss zwischen dem zeitlichen Gewinn und dem Verlust der Genauigkeit der Klassifikation abgewogen werden. Einige Ansätze können jedoch auch in einem erweiterbaren System gut eingesetzt werden. Ein an Pruning-Verfahren angelehnter Ansatz zur merkmalsbasierten Einschränkung des Suchraums wird in Abschnitt 6.5 näher untersucht.

5.3 Zusammenfassung

In diesem Abschnitt wurde die Erweiterbarkeit von Objekterkennungsverfahren auf mehreren verschiedenen Stufen untersucht und jeweils erweiterbare Lösungen vorgeschlagen.

In Abschnitt 5.1 wurden Klassifikationsmethoden auf deren Erweiterbarkeit untersucht, wobei sich nichtparametrische Verfahren, insbesondere NN-basierte Ansätze als beste Lösung herausstellten. Auch aus Sicht der Skalierbarkeit wurde diese Entscheidung basierend auf den Erkenntnissen aus [BSI08] und [DBLFF10] unterstützt.

Des Weiteren wurde in Abschnitt 5.1.1 die Erstellung von visuellen Vokabularen bei BoW-Ansätzen untersucht und eine erweiterbare Lösung aufbauend auf klassenspezifischen Vokabularen vorgeschlagen. Weitere Merkmale für den Einsatz in einem erweiterbaren Objekterkennungssystem wurden in Abschnitt 5.1.2 und Abschnitt 5.1.3 besprochen. Als sinnvoll wird der Einsatz von GIST und BoC erachtet.

Für die möglichst optimale Gewichtung und Kombination der Merkmale wurden in Abschnitt 5.1.4 verschiedene Möglichkeiten vorgestellt, aus denen der random-restart Bergsteigeralgorithmus zur weiteren Verwendung ausgewählt wurde.

Verschiedene Methoden zur Realisierung einer effizienten Suche wurden in Abschnitt 5.2 bzgl. deren Einsatz in einem erweiterbaren Objekterkennungssystem untersucht. Im weiteren Verlauf wird ein Verfahren zur merkmalsbasierten Einschränkung des Suchraumes verfolgt.

Das vorgestellte erweiterbare Verfahren wird im folgenden Kapitel mittels verschiedener Datensätze optimiert.

KAPITEL 6

OPTIMIERUNG DES ERWEITERBAREN VERFAHRENS

In Kapitel 5 wurden verschiedene Aspekte der objekterkennungsbasierten Annotation von Bildern aus der Sicht der Erweiterbarkeit beleuchtet und Lösungen für die verschiedenen Stufen der Klassifikation ermittelt. Die erweiterbaren Lösungsansätze können durch verschiedene Parameter eingestellt werden, wobei es diejenige Konfiguration zu finden gilt, bei der die Annotation der Bilder am besten ausfällt. In der Literatur werden für die Bewertung der Objekterkennung die Maße AUC und MAP verwendet, so dass in diesem Kapitel für die Optimierung und Feinabstimmung des vorgeschlagenen erweiterbaren Ansatzes ebenfalls diese Bewertungsmaße herangezogen werden. Als Datensätze für die Evaluation werden Caltech 256, PASCAL VOC 2007 und ImageNet eingesetzt.

In Abschnitt 6.1 wird die bestmögliche Größe und Einschränkung des auf dem BoW-Ansatz aufbauenden klassenspezifischen Vokabulars bestimmt. Abschnitt 6.2 evaluiert die Verwendbarkeit des Szenendeskriptors GIST und vergleicht verschiedene Variationen von Klassenbeschreibungen. Für die Verwendung von Farben bei der Objekterkennung wird in Abschnitt 6.3 der BoC-Ansatz unter Verwendung von verschiedenen Paletten untersucht. Die optimale Kombination der verwendeten Merkmale wird in Abschnitt 6.4 ermittelt. Zuletzt werden in Abschnitt 6.5 die Auswirkungen durch die approximative NN-Suche basierend auf der merkmalsbasierten Einschränkung des Suchraums aus-

Abschnitt 5.2.2.2 beleuchtet. Die Ergebnisse aus diesem Kapitel werden in Abschnitt 6.6 zusammengefasst.

6.1 Klassenspezifische visuelle Vokabulare

Im Abschnitt 5.1.1 wurden die Probleme von visuellen Vokabularen des Bag-of-Words-Ansatzes bzgl. der Erweiterbarkeit beleuchtet. Ausgehend von der Bewertung der unterschiedlichen Lösungsmöglichkeiten in Abschnitt 5.1.1.3 wurde ein erweiterbarer Ansatz unter Verwendung von klassenspezifischen Vokabularen ausgearbeitet.

Im Folgenden werden die optimalen Parameter für die in Abschnitt 5.1.1.4 vorgestellte Methode ermittelt. Es wird untersucht, wie sich die Wahl der initialen Vokabulargröße k , die Einschränkung auf die häufigsten n visuellen Wörter und die Anzahl der Trainingsbilder auf die Klassifikation auswirken. Für den Vergleich werden die Bewertungsmaße AP und AUC verwendet.

Als Datensatz wurde Caltech 256 von [GHP07] eingesetzt. Da die Genauigkeit der Klassifikation auch abhängig von den für den Testlauf gewählten Klassen ist, zeigen die Auswertungen die Mittelwerte von 100 Testläufen, bei denen jeweils zufällig 3, 4, 5, 10 bzw. 20 verschiedene Klassen aus dem Caltech 256 Datensatz ausgewählt wurden. Somit kann ein Eindruck davon gewonnen werden, wie generisch die evaluierte Methode ist. Bei jedem Experiment wurden die Bilder – wegen der affinen Eigenschaft der eingesetzten Detektoren und Deskriptoren ohne Berücksichtigung des Seitenverhältnisses – auf 128x128 Pixel verkleinert, in den HSV-Farbraum konvertiert und der V-Kanal (Helligkeitsinformation) extrahiert. Anschließend wurden mit dem Hessian-Affine-Detektor interessante Punkte im Bild ermittelt und durch SIFT-Deskriptoren beschrieben. Zur Extraktion und Beschreibung der Punkte wurde die Software aus [MS05] und [MTS⁺05] verwendet.

6.1.1 Bestimmung der Vokabulargröße

In diesem Versuch wurde untersucht, wie sich die Genauigkeit der Klassifikation in Abhängigkeit von der Vokabulargröße k verändert. Das Verhältnis zwischen der Anzahl der Vokabulargröße k und der behaltenen häufigsten n visuellen Wörter wurde auf $n/k = 0,5$ festgesetzt, d. h. es wurde je Test die Hälfte der visuellen Wörter des Vokabulars als Beschreibung der Klasse abgespeichert. In den Experimenten wurden Vokabulargrößen von $k = 40, 60, 80$ und 100 eingesetzt. Größere Werte wurden für k nicht evaluiert, da

bei einer Bildgröße von 128x128 Pixel unter Verwendung des Hessian-Affine Detektors im Durchschnitt 275 interessante Punkte je Bild gefunden wurden. Die Ergebnisse sind in Abbildung 6.1 veranschaulicht.

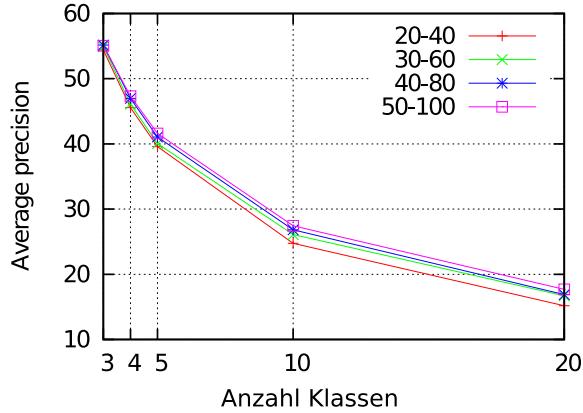


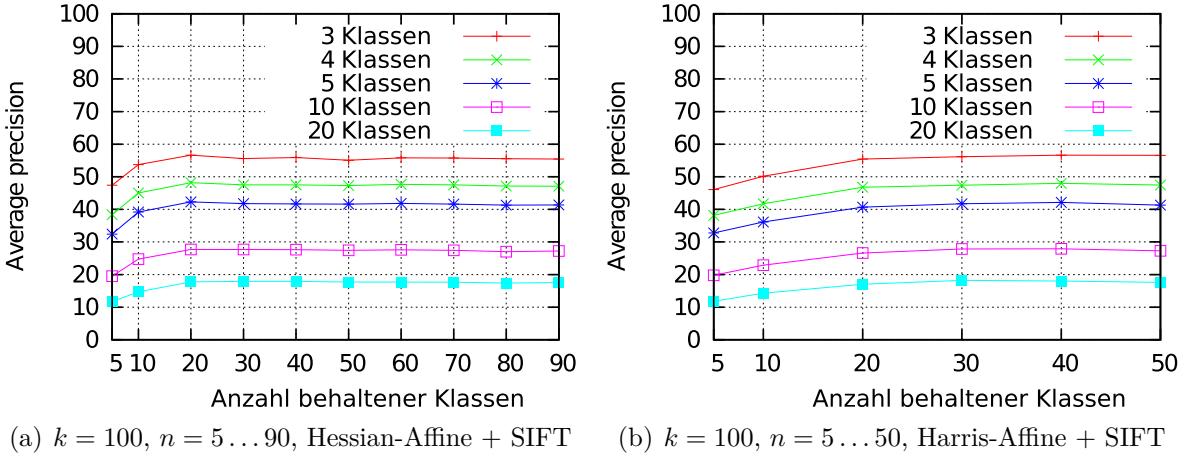
Bild 6.1: Vergleich von verschiedenen Vokabulargrößen $k = 40, 60, 80, 100$ unter Behaltung der Hälfte der häufigsten visuellen Wörter ($n/k = 0,5$).

Ähnlich zu den Ergebnissen aus [NS06] wurde auch bei den Tests mit klassenspezifischen Vokabularen bestätigt, dass je größer das Vokabular gewählt wird, desto besser auch die Genauigkeit der Klassifikation wird. Ebenfalls wichtig anzumerken ist, dass die Differenz zwischen den Genauigkeiten umso größer wird, je höher die Anzahl der Klassen ist.

6.1.2 Bestimmung der Schnittgrenze

Im nächsten Versuch wurde untersucht, ab wieviel Prozent der häufigsten visuellen Wörter eines klassenspezifischen Vokabulars sich die Genauigkeit der Klassifikation nicht weiter verändert. Basierend auf den Ergebnissen der Auswertung in Abbildung 6.1 wurde in diesem Experiment eine Vokabulargröße von $k = 100$ verwendet. Dabei wurden die $n = 5$ bis $n = 90$ häufigsten visuellen Wörter als Beschreibung der Klassen behalten. Die Ergebnisse der Evaluation bezüglich der AP gemittelt über 100 Testläufe sind in Abbildung 6.2(a) dargestellt.

Aus den Ergebnissen ist klar abzulesen, dass bereits 20% der häufigsten visuellen Wörter die gleiche Genauigkeit bzgl. der Klassifikation liefern wie Klassenbeschreibungen mit mehr Wörtern. Praktisch heißt das, dass bereits 1/5 der klassenspezifischen Vokabulare ausreichend ist, womit sich einerseits der Speicherbedarf, andererseits auch die Berechnungszeit erheblich verringert. Um zu zeigen, dass dieses Verhältnis auch bei anderen Detektoren besteht, wurde das Testszenario unter Verwendung des Harris-Affine-



(a) $k = 100$, $n = 5 \dots 90$, Hessian-Affine + SIFT (b) $k = 100$, $n = 5 \dots 50$, Harris-Affine + SIFT

Bild 6.2: Vergleich der Genauigkeit der Klassifikation mit unterschiedlicher Anzahl der behaltenen häufigsten visuellen Wörtern n unter Verwendung der Vokabulargröße $k = 100$ für alle klassenspezifische Vokabulare.

Detektors wiederholt. Die Ergebnisse dieser Evaluation sind in Abbildung 6.2(b) zu sehen. Auch bei der Verwendung des Harris-Affine-Detektors zeigt sich ein ähnliches Bild mit einer Grenze von ca. 20 bis 25% der visuellen Wörter.

6.1.3 Bestätigung der Outlier-Hypothese

In Abschnitt 5.1.1.4 wurde der Begriff von visuellen Outlier-Wörtern eingeführt, welche denjenigen visuellen Wörtern eines Bildes entsprechen, die keinem der n visuellen Wörter der Beschreibung einer Klasse zugeordnet werden konnten, da sie zu weit entfernt vom Clusterzentrum sind. Basierend auf dieser Definition wurde die Hypothese aufgestellt, dass die Anzahl der Outlier-Wörter bei einem Bild, welches zur richtigen Klasse gehört, im Durchschnitt niedriger sein wird, als bei einem Bild, welches zu einer anderen Klasse gehört. Zur Bestätigung dieser Hypothese wurde die durchschnittliche Anzahl der Outlier-Wörter bei verschiedenen Vokabulargrößen berechnet. Die Ergebnisse in Abbildung 6.3 belegen die Annahme bzgl. der Outlier-Wörter.

6.1.4 Bewertung

Für die allgemeine Bewertung der Klassifikation wurden die ROC-Kurven für die Vokabulargröße $k = 100$ und Beibehaltung der $n = 5$ bis $n = 50$ häufigsten visuellen Wörter berechnet. Die Fläche unter der besten ROC-Kurve (AUC) in Abbildung 6.4 beträgt

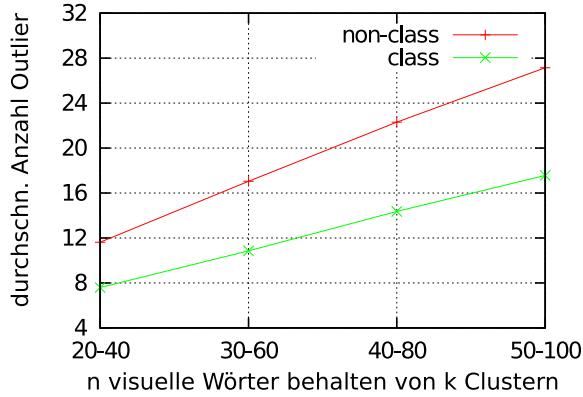


Bild 6.3: Durchschnittliche Anzahl der visuellen Outlier-Wörter für Bilder, welche zur richtigen, und Bilder, welche nicht zur gegebenen Klasse gehören.

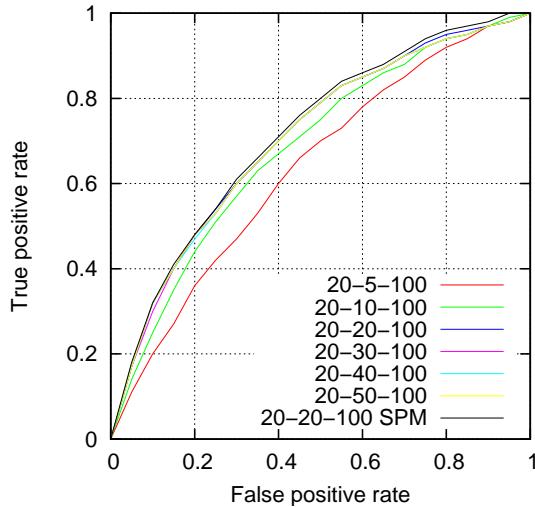


Bild 6.4: ROC-Kurven für Vokabulargröße $k = 100$ mit verschiedenen Werten für die Anzahl der behaltenen häufigsten visuellen Wörter n bei 20 Trainingsbildern je Klasse.

0,72. Durch den Einsatz von Spatial Pyramids gemäß [LSP06] konnte dieser Wert auf 0,75 verbessert werden, was jedoch die Berechnungszeit erheblich ansteigen ließ.

Zusätzlich wurde die Skalierbarkeit für eine große Anzahl von Klassen untersucht. Hierzu wurden die 3 251 durch Bounding Boxes annotierten Klassen von ImageNet verwendet. Für die einzelnen Testdurchläufe wurde die Menge aller Klassen auf Gruppen der Größe N aufgeteilt, die MAP berechnet und die MAP über alle Gruppen hinweg gemittelt. Somit spiegelt die Evaluation die MAP unabhängig von der Konstellation der Gruppen wider.

Für die Bewertung wurden 20 Trainings- und Testbilder zufällig ausgewählt. Die Bilder wurden auf 128x128 Pixel skaliert, mittels dem Hessian-Affine-Detektor und durch dense Sampling interessante Punkte ermittelt und durch SIFT-Deskriptoren beschrieben. Aus den SIFT-Deskriptoren der Trainingsbilder wurden anschließend klassenspezifische Vokabulare mit $k = 100$ Wörtern berechnet, wovon die häufigsten $n = 20$ behalten wurden. Die Ergebnisse sind in Abbildung 6.5 dargestellt.

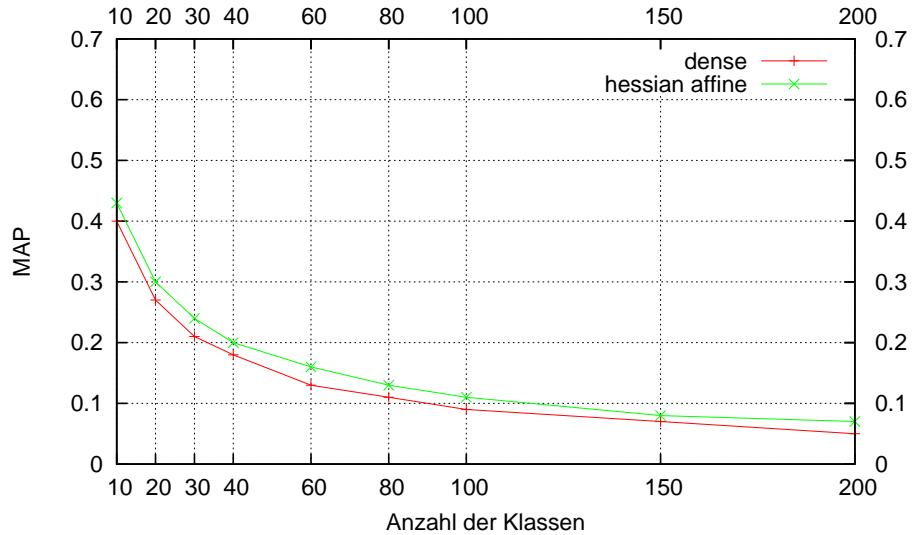


Bild 6.5: Vergleich der MAP von klassenspezifischen BoW-basierten Ansätzen unter Verwendung von verschiedenen Detektoren bei steigender Anzahl von Klassen.

Die Evaluation zeigt ähnlich zu [DHS00] eine stetige Abnahme der MAP bei steigender Anzahl von Klassen, wobei für eine hohe Anzahl von Klassen die Kurve langsam abflacht. Sowohl die auf dense Sampling als auch die auf dem Hessian-Affine-Detektor basierenden Verfahren liegen dicht beieinander, wobei Letzteres in allen Fällen leicht besser abschneidet. Da in der Literatur allgemein dense und sparse Sampling kombiniert Verwendung finden, werden später im Annotationsframework beide Detektorvarianten eingesetzt.

6.1.5 Zusammenfassung

In den vorgestellten Auswertungen konnte ermittelt werden, dass je größer das initiale Vokabular gewählt wird, desto besser die Klassifikation. Eine interessante Erkenntnis war, dass bereits 20 bis 25% der häufigsten visuellen Wörter des klassenspezifischen Vokabulars die gleiche Genauigkeit bzgl. der Klassifikation aufzeigen wie größere Klas-

senbeschreibungen. Dadurch kann erheblicher Speicherplatz und Rechenzeit eingespart werden. Die Hypothese bzgl. der Anzahl von visuellen Outlier-Wörtern konnte belegt werden. Bilder, welche zu einer gegebenen Klasse gehören, beinhalten im Durchschnitt wesentlich weniger Outlier-Wörter als Bilder die zu anderen Klassen gehören. Auch die Skalierbarkeit des Ansatzes wurde evaluiert, wobei festgestellt wurde, dass die Abnahme der MAP bei einer hohen Anzahl von Klassen langsam abflacht.

6.2 Szenendeskriptor für Objekte

In Abschnitt 5.1.2 wurde ein Verfahren zum Einsatz des GIST-Szenendeskriptors für die Objekterkennung beschrieben. Für die Evaluation des Verfahrens wurde der PASCAL-VOC-2007-Datensatz von [EVGW⁺07] verwendet, da dieser Datensatz bereits Rahmen um die in den Bildern enthaltenen Objekten enthält inkl. der Annotation, um welches Objekt es sich handelt, sowie dessen Ausrichtung. Zwar sind auch neuere PASCAL-VOC-Datensätze verfügbar, für den Vergleich von Merkmalen wird jedoch der Datensatz aus dem Jahr 2007 empfohlen, da bei diesem auch die Testdaten öffentlich zugänglich gemacht wurden.

Trotz der vorhandenen Annotation im PASCAL-VOC-2007-Datensatz fehlten bei vielen Objekten Informationen über deren Ausrichtung. Diese wurden zusätzlich manuell annotiert sowie fehlerhafte Annotationen bereinigt. Somit wurden die 20 Basisklassen des Datensatzes auf insgesamt 154 Subklassen aufgeteilt. Einige Subklassen wurden ausgelassen, da sie die festgelegte minimale Anzahl von 9 Bildern je Subklasse in der Trainings- bzw. Testmenge nicht erreichten. Als „schwierig“ eingestufte Bilder wurden ebenfalls nicht betrachtet, da sie in den PASCAL-Wettbewerben nicht berücksichtigt werden. Jedes Teilbild wurde unabhängig vom Seitenverhältnis auf 32x32 Pixel herunterskaliert. Anschließend wurden 960-dimensionale GIST-Deskriptoren für jedes Teilbild mittels der GIST-Implementierung von [DJS⁺09] berechnet.

Da jede Subklasse in der Testmenge mindestens 9 Bilder enthält, wurden für die Tests je 9 Bilder pro Subklasse per Zufall aus den Testbildern bestimmt. Bei der Evaluation wurden mittels der L1- und L2-Distanzfunktionen die Abstände zwischen GIST-Deskriptoren berechnet. Ein Bild wurde als korrekt klassifiziert eingestuft, wenn es einer Subklasse der übergeordneten Klasse zugeordnet wurde. Wenn also ein Bild, welches ein Motorrad von vorne zeigt, z. B. der Subklasse Motorrad_{links} zugeordnet wird, ist die Klassifikation

korrekt, da beide Subklassen zu der übergeordneten Klasse Motorrad gehören. Als Bewertungsmaße wurden bei der Evaluation AUC, AP und MAP verwendet.

6.2.1 Vergleich mit klassenspezifischen Vokabularen

Im ersten Szenario wurde das GIST-basierte Verfahren mit dem erweiterbaren BoW-basierten Ansatz aus Abschnitt 5.1.1.2 verglichen. Die Berechnung der klassenspezifischen Vokabulare erfolgte ähnlich wie in Abschnitt 6.1 beschrieben, mit dem Unterschied, dass ein Vokabular je Subklasse basierend auf je 9 Trainingsbildern erstellt wurde. Für die GIST-Varianten wurden ebenfalls 9 Trainingsbilder je Subklasse verwendet.

Neben der NN-Suche auf allen Deskriptoren der Trainingsbilder wurden zusätzlich zwei Reduktionsmaßnahmen evaluiert. Bei meanGIST wurde nur der Mittelwert, bei medianGIST nur der Median der GIST-Deskriptoren der Trainingsbilder je Subklasse abgespeichert und für die NN-Suche herangezogen. Das führte zu einem geringeren Speicherbedarf, weniger Vergleichsoperationen und dadurch insgesamt zu einer schnelleren Antwortzeit. Die Ergebnisse der Messungen sind als AUC-Werte in Tabelle 6.1 aufgeführt. Das Verhältnis der Ausführungszeiten ist ebenfalls in der Tabelle vermerkt, wobei die Zeit von meanGIST als Referenzwert genommen wurde. Die Tests wurden mehrfach mit verschiedenen zufällig ausgewählten Trainingsbildern wiederholt. In der Tabelle 6.1 ist jeweils der beste erreichte und der durchschnittliche AUC-Wert enthalten.

	cs-BoW	NNIST	MEDIANGIST	MEANGIST
Durchschn. AUC L1	0.7315	0.8251	0.7885	0.8471
Durchschn. AUC L2	0.7373	0.8176	0.7879	0.8428
Beste AUC L1	0.7493	0.8560	0.8191	0.8768
Beste AUC L2	0.7515	0.8463	0.8184	0.8762
Durchschn. Ausführungszeit	27.8	8.7	1	1

Tabelle 6.1: Vergleich der AUC-Werte des erweiterbaren BoW-Ansatzes aus Abschnitt 5.1.1.4 (Spalte *cs-BoW*) und der GIST-basierten Ansätze unter Verwendung von blickwinkelbasierten Subklassen mit jeweils 9 Trainingsbildern. Die letzte Zeile der Tabelle veranschaulicht das Verhältnis zwischen den durchschnittlichen Ausführungszeiten, wobei als Referenz meanGIST genommen wurde.

Die Evaluation zeigt, dass bei der Verwendung von Subklassen mit wenigen (9) Trainingsbildern alle GIST-basierten Ansätze besser abschneiden als klassenspezifische Voka-

bulare. Der Unterschied zwischen der L1- und L2-Distanz ist vernachlässigbar gering. Auffällig ist jedoch, dass meanGIST sowohl im Durchschnitt als auch beim besten AUC vorne liegt. Die Gründe für das gute Abschneiden von meanGIST werden bei der detaillierten Evaluation in Abschnitt 6.2.4 besprochen, da erst durch die Betrachtung der AP-Werte für einzelne Klassen sich eine genaue Erklärung hierfür erschlossen werden kann.

6.2.2 Abhängigkeit von der Anzahl der Trainingsbilder

Im zweiten Szenario wurden die verschiedenen GIST Ansätze mit unterschiedlich vielen Trainingsbildern evaluiert. Dabei wurden je Subklasse 1, 3, 5 und 9 Trainingsbilder zufällig ausgewählt. Die Tests wurden mehrfach mit unterschiedlichen Trainingsbildern wiederholt. Als Bewertungsmaße wurden AUC und MAP verwendet.

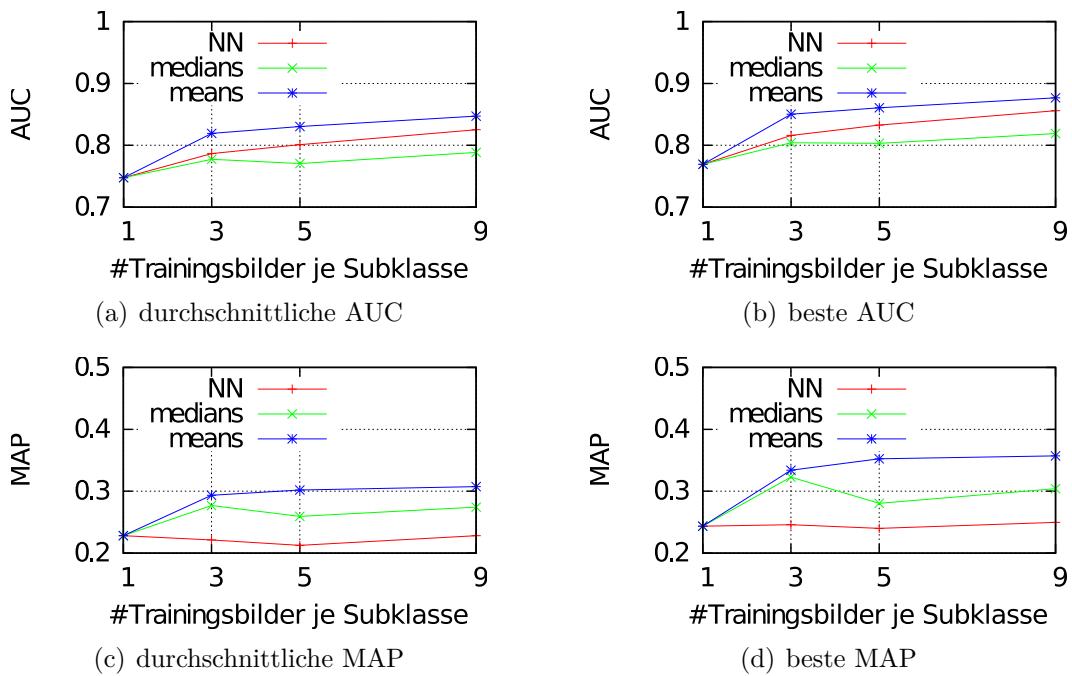


Bild 6.6: Vergleich der besten und durchschnittlichen AUC- und MAP-Werte für die verschiedenen GIST-basierten Verfahren mit unterschiedlicher Anzahl von Trainingsbildern.

Es wurde zwar vermutet, aber aus den Bildern ist auch ersichtlich, dass bei einer größeren Zahl an Trainingsbildern sich die Klassifikation verbessert. Verwunderlich ist jedoch, dass bereits bei der Verwendung von 3 Trainingsbildern je Subklasse meanGIST

sehr gute Ergebnisse liefert. Zusätzlich kann festgestellt werden, dass bei allen Testläufen meanGIST sowohl bzgl. AUC als auch MAP am besten abschnitt.

6.2.3 Rotationsinvarianz

Alle annotierten Objektrahmen im PASCAL-VOC-2007-Datensatz sind parallel zu den Seiten des Bildes. Dadurch könnten sich bei verdrehten Objekten in den Teilbildern Verschiebungen ergeben, welche meanGIST ungünstig beeinflussen. Zur Evaluation dieses Problems wurde der Einsatz von rotationsinvarianten Objektrahmen untersucht.

Um Rotationsinvarianz auf den Teilbildern zu erreichen, wurde das einfache Verfahren aus [BETVG08] verwendet, welches auch für die Implementierung des SURF-Deskriptors eingesetzt wurde. Dazu wurden Haar-Wavelets als Filter auf das verkleinerte Bild in x - und y -Richtungen berechnet, anschließend in $\frac{\pi}{3}$ -Schritten aufsummiert und der längste Vektor als Hauptachse bestimmt. Die Teilbilder wurden darauffolgend basierend auf dieser berechneten Hauptachse gedreht.

Im Vergleich zu nicht rotierten Teilbildern wurde nach der Anwendung der oben beschriebenen Rotationsinvarianz bei meanGIST leider ein Verlust von mehr als 50% bzgl. der MAP festgestellt. Eine Verbesserung ergab sich lediglich bei freischwebenden Objekten mit einfacher Struktur, wie z. B. bei Flugzeugen. Die meisten Objekte im PASCAL-VOC-2007-Datensatz weisen entweder eine wesentlich komplexere Struktur auf, bei der nicht eindeutig eine Hauptachse für die korrekte Rotation feststellbar ist, oder sie sind bereits ganz natürlich durch die Aufnahme des Fotos in die richtige Position gedreht (z. B. Fahrrad steht auf dem Boden). Bei beiden genannten Eigenschaften wird durch die Anwendung der vorgestellten Methode zur Rotationsinvarianz eher Rauschen in die Daten hineingebracht als entfernt.

6.2.4 Unterschiede auf Klassenebene

In diesem Szenario wurde der GIST-basierte Ansatz mit den besten und durchschnittlichen MAP-Werten des PASCAL-VOC-Wettbewerbs aus 2007 verglichen. Für die GIST-basierten Ansätze wurden je Subklasse 9 Trainingsbilder verwendet, was deutlich weniger ist als bei den Teilnehmern des Wettbewerbs. In der Tabelle 6.2 sind für die GIST-Verfahren die durchschnittlichen AP-Werte je Klasse sowie der durchschnittliche MAP-Wert aufgeführt.

KLASSE	BEST	MEAN	MEANGIST	NNGIST
aeroplane	0.775	0.651	0.259	0.299
bicycle	0.636	0.462	0.659	0.237
bird	0.561	0.387	0.101	0.131
boat	0.719	0.541	0.452	0.371
bottle	0.331	0.219	0.049	0.088
bus	0.606	0.432	0.411	0.132
car	0.780	0.649	0.391	0.252
cat	0.588	0.434	0.136	0.211
chair	0.535	0.449	0.185	0.164
cow	0.426	0.277	0.273	0.113
diningtable	0.549	0.377	0.069	0.144
dog	0.458	0.367	0.109	0.171
horse	0.775	0.657	0.297	0.108
motorbike	0.640	0.506	0.707	0.262
person	0.859	0.784	0.284	0.227
pottedplant	0.363	0.245	0.220	0.131
sheep	0.447	0.304	0.446	0.305
sofa	0.509	0.353	0.095	0.127
train	0.792	0.622	0.363	0.089
tvmonitor	0.532	0.400	0.644	0.463
MAP	0.575	0.427	0.308	0.201

Tabelle 6.2: Vergleich der AP der besten und Durchschnitts-AP-Werte des PASCAL VOC 2007 Wettbewerbs mit dem vorgestellten meanGIST- und NNGIST-Ansatz. Die Trainingsmenge bestand aus 9 Bildern je Subklasse. Zur Berechnung der Abstände wurde die L1-Distanz eingesetzt.

Für einige Klassen, wie z. B. Fahrrad, Motorrad oder Fernseher, ergeben sich bei meanGIST ähnlich gute oder sogar bessere AP-Werte als bei dem besten Verfahren des PASCAL-VOC-Wettbewerbs aus 2007. Andere Klassen jedoch, wie z. B. Hund, Katze oder Pferd, schneiden schlechter ab. Als Grund dafür wird vermutet, dass letztere Klassen innerhalb einer Subklasse viele verschiedene Stellungen einnehmen können, welche ein zu starkes Rauschen verursachen. Zum Beispiel eine Katze von links abgebildet kann in verschiedene Richtungen schauen und dabei liegen, sitzen oder stehen. Daraus lässt sich ableiten, dass GIST für Objekte mit hohen Variationen weniger gut geeignet ist, jedoch sehr gut bei Objekten mit niedrigen Variationen funktioniert.

Abbildung 6.7 zeigt Durchschnittsbilder für Klassen mit hohen (Fahrrad in Abbildung 6.7(a) und Motorrad Abbildung 6.7(b)) und niedrigen Werten für AP (Katze

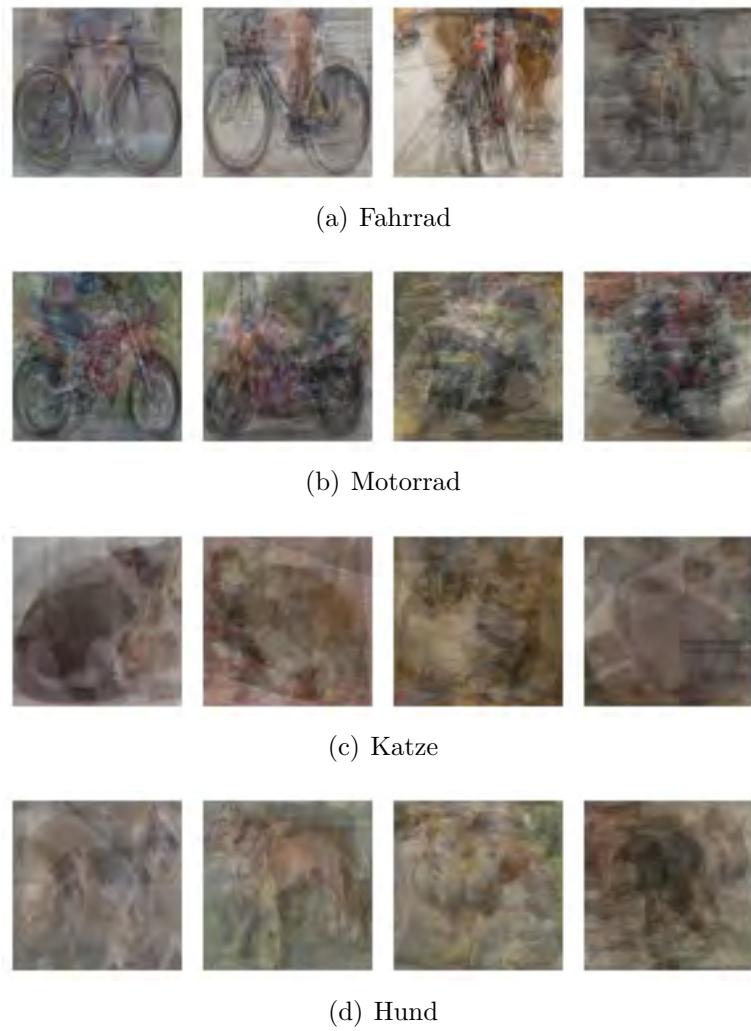


Bild 6.7: Durchschnittsbilder erstellt aus 9 Trainingsbildern für die Subklassen *rechts, links, vorne* und *hinten* für die Klassen *Fahrrad, Motorrad, Katze* und *Hund*.

Abbildung 6.7(c) und Hund Abbildung 6.7(d)). Die Durchschnittsbilder entsprechen ungefähr der Darstellung der meanGIST-Beschreibungen der einzelnen blickwinkelbasierten Subklassen.

Bei den Durchschnittsbildern der Motorräder und Fahrräder ist die räumliche Struktur der Objekte immer noch gut erkennbar. Die Durchschnittsbildung hat sogar den Vorteil, dass sie kleine Verschiebungen bei den einzelnen Objekt elementen erlaubt, dabei aber sowohl die räumliche Struktur als auch das Verhältnis des Vorder- und Hintergrunds beibehält. Im Gegensatz dazu erscheinen Hunde und Katzen ganz natürlich in vielen verschiedenen Variationen und Stellungen. Daraus folgt auch, dass deren Durchschnitts-

bilder viel verschmierter aussehen, was dazu führt, dass die räumliche Struktur der Szene und das Verhältnis von Vorder- und Hintergrund verloren geht. Einerseits beantwortet diese Beobachtung die Frage, warum bei der Verwendung von GIST einige Klassen wesentlich schlechter abschneiden als andere. Andererseits wird dadurch auch erklärt, warum meanGIST für Klassen mit niedrigen Variationen und NNGIST bei Klassen mit hohen Variationen bessere AP-Werte liefert. Basierend auf diesen Erkenntnissen könnte ein Vorverarbeitungsschritt eingefügt werden, welcher den Grad der Verschmiertheit des Durchschnittsbildes einer Subklasse ermittelt und aufgrund dieser Information meanGIST oder NNGIST für die Klassifikation verwendet.

6.2.5 Vollständig automatische Annotation

In diesem Szenario wurde der Einsatz des GIST-Verfahrens in einer durchgängig vollautomatischen Annotation evaluiert. Dazu wurde das Bild zuerst mittels der konturbasierten Methode von [AMFM09] segmentiert und zusammenhängende Bereiche durch seitenparallele Vierecke umrahmt. Die so entstandenen Teilbilder wurden anschließend klassifiziert und durch die entsprechende Bezeichnung der am nächsten gelegenen Klasse annotiert. Die Ergebnisse für 4 ausgewählte Bilder des PASCAL-VOC-2007-Datensatzes sind in Abbildung 6.8 dargestellt. Neben dem Originalbild ist die Segmentierung mittels dem Verfahren von [AMFM09], die entstandene Umrahmung sowie die Position der korrekten Klasse für das gegebene Teilbild zu sehen.

Die Ergebnisse zeigen, dass die Verwendung von GIST auf Objekte in Verbindung mit einer vollautomatischen Segmentierung gute Annotationen liefert. Diejenigen Fälle, bei denen die korrekte Klasse nicht an erster Stelle stand, wurden näher untersucht. Beim Foto vom Pferd stellte sich heraus, dass es oft als Kuh klassifiziert wurde. Das kann dadurch begründet werden, dass sowohl die Trainings- als auch die Testbilder ohne Berücksichtigung der Seitenverhältnisse auf 32x32 Pixel verkleinert wurden. Dies hatte zur Folge, dass die längliche Natur bei von vorne abgebildeten Pferden verloren ging und somit einer Kuh recht nahe kam.

Das letzte Beispielbild enthält einen Sessel, welcher im PASCAL VOC 2007 Datensatz durchweg unter der Klasse Stuhl eingeordnet wurde, allerdings auch als ein enges Sofa angesehen werden kann. Hier ist der Grund für die Klassifikation der Sessel als Sofa ebenfalls durch die Missachtung der Seitenverhältnisse bei der Skalierung der Bilder auf 32x32 Pixel begründbar. Ein Foto von einem Sofa, welches in Quadratform gedrückt

6. Optimierung des erweiterbaren Verfahrens



Bild 6.8: Ergebnisse der vollautomatischen Annotation von 4 ausgewählten Bildern des PASCAL VOC 2007 Datensatzes. Neben den Originalbildern ist die Segmentierung mittels des Verfahrens nach [AMFM09] zu sehen. Im Bild daneben wurden die zusammenhängenden Bereiche durch seitenparallele Vierecke umrahmt. Die so entstandenen Teilbilder wurden schließlich durch verschiedene GIST-basierte Ansätze klassifiziert. Die Ergebnisse zeigen die Position der korrekten Klasse bei der Klassifikation.

wird, ist mit dem Bild von einem Sessel quasi identisch. Generell könnte in beiden geschilderten Problemfällen die Berücksichtigung des Seitenverhältnisses der Teilbilder ggf. die Genauigkeit verbessern.

6.2.6 Skalierbarkeit

Ähnlich zum Abschnitt 6.1.4 wurden zur Untersuchung der Skalierbarkeit GIST-Deskriptoren unter Verwendung von unterschiedlich vielen Klassen evaluiert. Als Daten-

satz wurden die 3 251 mit Bounding Boxes annotierten Klassen von ImageNet verwendet. Es wurden jeweils Gruppen der Größe N gebildet, die MAP berechnet und die MAP für alle Gruppen der gleichen Größe gemittelt. Somit zeigt die Evaluation die MAP unabhängig von der Zusammensetzung der Klassen bei den einzelnen Testfällen. Die Ergebnisse sind in Abbildung 6.9 dargestellt.

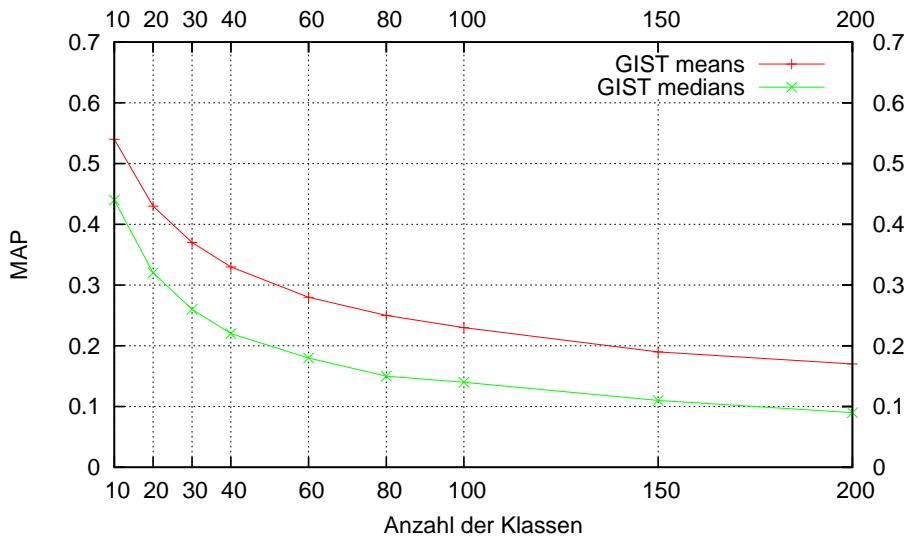


Bild 6.9: Vergleich der MAP von verschiedenen GIST-basierten Ansätzen bei unterschiedlichen Anzahlen von Klassen.

Ähnlich zu den Kurven in Abbildung 6.5 ist auch hier erwartungsgemäß ein stetiger Abfall der MAP bei steigender Anzahl von Klassen zu beobachten, wobei die MAP generell höher ausfällt als in Abbildung 6.5. Des Weiteren zeigt die Evaluation einen deutlichen Unterschied zwischen den beiden GIST-basierten Verfahren. Aus diesem Grund werden im Annotationsframework die klassenspezifischen durchschnittlichen GIST-Deskriptoren eingesetzt.

6.2.7 Zusammenfassung

In diesem Abschnitt wurde der eigentlich für die Erkennung von Szenen entwickelte GIST-Deskriptor für den Einsatz in der Objekterkennung evaluiert. Neben der NN-Suche direkt auf den GIST-Deskriptoren der Trainingsbilder wurde auch die klassenbezogene Komprimierung der GIST-Deskriptoren untersucht. Dabei stellte sich meanGIST als beste und schnellste Variante heraus. Auch in Kombination mit einem aktuellen automatischen Segmentierungsverfahren schnitt der GIST-Deskriptor sehr gut bei der Erkennung von

Objekten im PASCAL-VOC-2007-Datensatz ab. Zusätzlich wurde auch die Skalierbarkeit auf 3251 Klassen des ImageNet-Datensatzes evaluiert. GIST ist sehr kompakt und liefert gleichzeitig sehr gute Erkennungsraten.

6.3 Bag of Colors

In Abschnitt 5.1.3.2 wurde das BoC-Verfahren zur Erkennung von Objekten beschrieben. Der Ansatz vereinigt im Wesentlichen die Ideen von BoW und Farbhistogrammen.

In [WDJ11] wurde gezeigt, dass globale BoC-Histogramme bei der Objekterkennung besser abschneiden als BoW mit SIFT oder Farbvarianten von SIFT-Deskriptoren. Es wurde ebenfalls nachgewiesen, dass je mehr Farben, also je mehr Dimensionen das BoC-Histogramm besitzt, desto besser die MAP ist. Aufbauend auf diesen Ergebnissen wurde in der vorliegenden Arbeit eine Farbpalette mit 256 Farben verwendet.

Im Nachfolgenden wurde die Auswirkung der Wahl der Farbpalette evaluiert. Zwar wurde eine ähnliche Untersuchung bereits in [WDJ11] durchgeführt, sie befasste sich jedoch nicht mit der Skalierbarkeit bzgl. einer großen Anzahl an Klassen. Für die Durchführung der Evaluation der Skalierbarkeit wurde im ersten Schritt eine Farbpalette aus 10 000 zufällig ausgewählten Bildern aus dem ImageNet Datensatz erstellt. Die Erstellung der Farbpalette folgte der Beschreibung von [WDJ11] und ist auch an die Erstellung des visuellen Vokabulars aus dem BoW-Ansatz angelehnt. Die Bilder wurden zuerst auf 256x256 Pixel verkleinert, in den $L^*a^*b^*$ -Farbraum konvertiert und je 16x16 Pixel-block die am häufigsten auftretende Farbe ermittelt. Die so erhaltenen Farben wurden anschließend mittels k -means auf die Vokabulargröße (Palettengröße) 256 geclustert.

Anschließend wurden mittels dieses Farbvokabulars BoC-Histogramme für alle Bilder erstellt. Als Vergleich dienten Histogramme, die mittels einer gleichverteilten Farbpalette ermittelt wurden. Zur Evaluation wurde die durch Bounding Boxes annotierte Teilmenge des ImageNet-Datensatzes verwendet. Für jede der mehr als 3 000 Klassen wurden 20 Trainings- und Testbilder per Zufall ausgewählt. Aus den Histogrammen der Trainingsbilder wurde ähnlich wie auch beim GIST-Deskriptor in Abschnitt 6.2 der Durchschnittswert und der Median berechnet. Diese Durchschnittshistogramme dienten als Beschreibung einer Klasse, mit denen die Testbilder mittels der L1-Distanz verglichen wurden.

Um die Skalierbarkeit des BoC-Ansatzes zu bewerten, wurden bei der Evaluation die 3251 Klassen in Gruppen der Größe N aufgeteilt, für die jeweils die MAP berechnet wurde. Anschließend wurde für alle Gruppen der Größe N der Mittelwert der MAP-Werte berechnet. Somit ergibt sich ein realitätsnaher MAP-Wert, der unabhängig von der aktuellen Konstellation der Gruppe ist. Abbildung 6.10 zeigt den Vergleich zwischen einer erlernten und gleichverteilten Farbpalette sowie die durchschnittlichen MAP-Werte für Gruppen unterschiedlicher Größen.

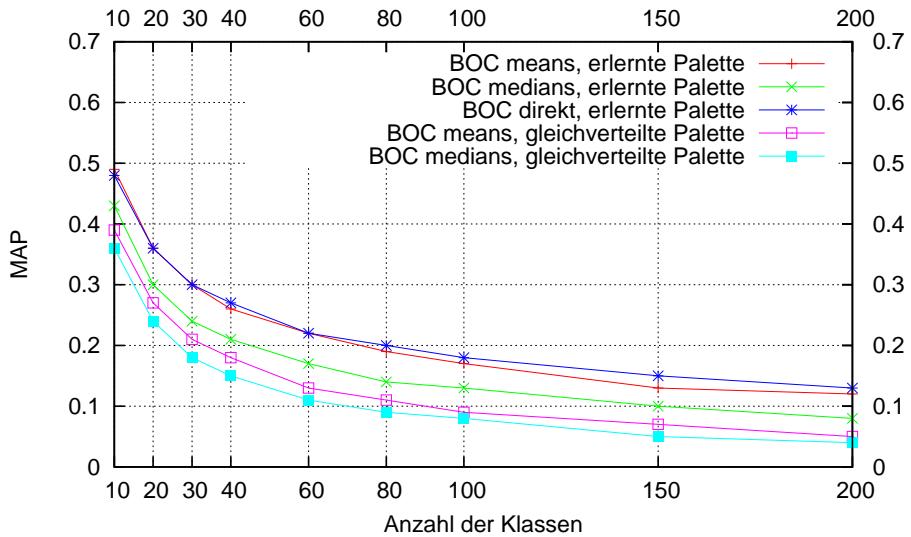


Bild 6.10: Vergleich der MAP von verschiedenen Bag of Colors Ansätzen bei unterschiedlichen Anzahlen von Klassen.

Die Evaluation zeigt, dass Histogramme basierend auf erlernten Farbpaletten wesentlich besser abschneiden als gewöhnliche Farbhistogramme. Außerdem ist es ersichtlich, dass die NN-Suche direkt auf den Histogrammen und die Suche auf den klassenspezifischen Durchschnittshistogrammen die beste MAP liefern. Beide liegen dicht beieinander, wobei in den Fällen mit vielen Klassen die direkte NN-Suche leicht besser abschneidet. Da allerdings die Differenz recht klein ist und die direkte NN-Suche 20 Mal (entspricht der Anzahl der Trainingsbilder) so viel Vergleiche und somit Speicherplatz und Zeit benötigt, werden für das Annotationsframework die klassenspezifischen Durchschnittshistogramme verwendet.

6.4 Kombination der Merkmale

In diesem Kapitel wurden für die in Kapitel 5 vorgestellten Merkmale optimale Parameter bestimmt und anhand der Datensätze Caltech 256 [GHP07], PASCAL VOC 2007 [EVGW⁺07] bzw. ImageNet [DBLFF10] einzeln evaluiert. Für die einzelnen Klassen können die verschiedenen Deskriptoren auch kombiniert werden, um die Genauigkeit der Objekterkennung zu erhöhen.

Für die Ermittlung einer möglichst optimalen Gewichtung bei der Kombination der vorgestellten Merkmale wurde das random-restart hill-climbing Verfahren verwendet. Dieser Algorithmus wurde bereits in [GMS08] für die Objekterkennung erfolgreich eingesetzt. Der Algorithmus zeichnet sich dadurch aus, dass er recht schnell die Wertelandschaft durchforstet und somit innerhalb weniger Iterationen bereits brauchbare Gewichtungen entstehen. Ein weiterer Grund war auch die relativ lange Zeit für die Berechnung der MAP-Werte für einzelne Zustände bei 3 251 Klassen.

In Abbildung 6.11 sind die MAP-Werte für die eingesetzten Merkmale jeweils einzeln sowie für die Kombination der Merkmale mit der ermittelten optimalen Gewichtung dargestellt. Die wesentliche Verbesserung durch die Kombination der Merkmale verdeutlicht, dass die jeweiligen Merkmale unterschiedliche Informationen der Bilder beschreiben.

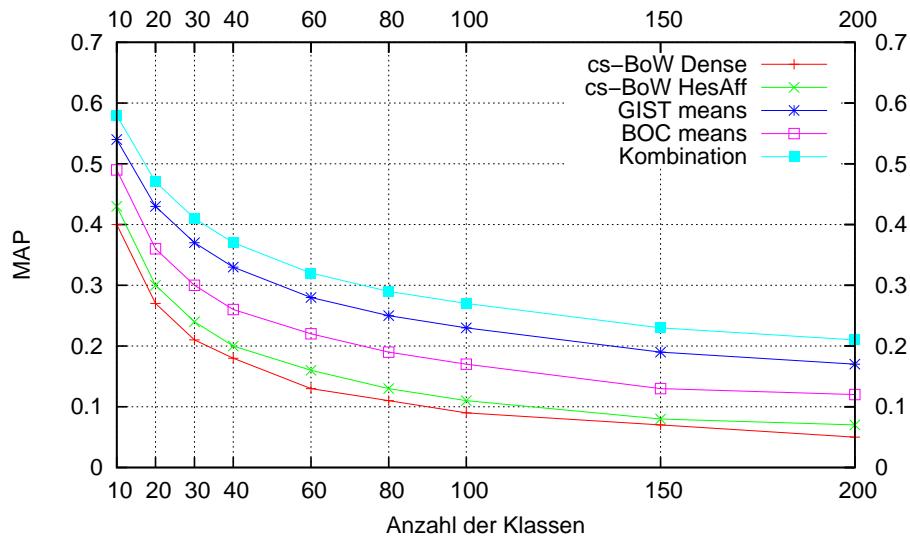


Bild 6.11: Vergleich der MAP für die einzelnen Merkmale und deren kombinierte Verwendung.

In Tabelle 6.3 sind die MAP-Werte für die eingesetzten Merkmale jeweils einzeln sowie für die gewichtete Kombination der Merkmale bei der Verwendung von allen 3 251 Klassen aufgeführt. Anhand dieser Werte und der in [DBLFF10] publizierten MAP-Werte kann festgestellt werden, dass die hier vorgestellte Kombination von Merkmalen für mehr als 3 000 Klassen sich im Bereich des aktuellen Stands der Technik befindet. Der Vorteil der vorliegenden Lösung im Vergleich zu den sonst üblichen SVM-basierten Lösungen ist die leichte und schnelle Erweiterbarkeit durch neue Klassen.

6.5 Einschränkung des Suchraums

In Abschnitt 5.2.2.2 wurde ein merkmalsbasiertes Pruning-Verfahren vorgestellt, mit dem für ein gegebenes Bild die Menge der irrelevanten Klassen ausgefiltert und somit die Berechnung mit weiteren Merkmalen verschnellert werden soll.

Die Genauigkeit der Approximierung der Menge von relevanten Klassen für die Annotation eines gegebenen Bildes hängt von mehreren Faktoren ab. Einerseits ist es wichtig, dass das ausgewählte Merkmal bzw. die Menge von Merkmalen für den Vergleich mit allen dem System bekannten Klassen schnell zu berechnen ist. Andererseits ist eine möglichst genaue Approximierung der eigentlichen Top- N -Nachbarn bei der Verwendung aller Merkmale und die damit verbundenen geringen Einbußen in der Annotation wünschenswert. In den nachfolgenden Abschnitten werden die oben genannten Faktoren für jedes im System verwendete Merkmal untersucht, um das möglichst beste Merkmal für die Vorfilterung der Klassenmenge zu finden.

MERKMAL	MAP
Bag of Colors	0,0387
cs-BoW (Dense Sampling)	0,0119
cs-BoW (Hessian-Affine)	0,0158
GIST	0,0653
Kombination	0,0830

Tabelle 6.3: Übersicht der MAP für die einzelnen Merkmale und deren Kombination bei 3 251 Klassen.

6.5.1 Durchschnittliche Berechnungszeiten je Merkmal

Durch die Vorfilterung der Klassenmenge soll die Annotation von Bildern schneller erfolgen. Daher ist es nötig, dass diese Vorverarbeitung mit möglichst geringem Aufwand und schnell erfolgt. Um ein Merkmal für die Vorfilterung auszuwählen, werden zuerst die durchschnittlichen Berechnungszeiten aller verwendeten Merkmale jeweils für die Klassifikation von Bildern ermittelt.

Abbildung 6.12 veranschaulicht die Laufzeiten je Merkmal für die Klassifikation bei der Verwendung von 3 251 Klassen. Die Laufzeiten stellen dabei den Mittelwert der Klassifikation von 100 zufällig ausgewählten Bildern dar. Die Verkleinerung des zu annotierenden Bildes, die Umrechnung des Farbraums und die Ermittlung der Merkmale wurde bei den Laufzeiten nicht berücksichtigt. Letzteres wird deswegen nicht zur Laufzeit hinzugerechnet, da alle Merkmale des zu annotierenden Bildes für die Klassifikation auf der eingeschränkten Klassenmenge benötigt werden. Die Laufzeiten in Abbildung 6.12 umfassen somit lediglich die Vergleiche der Merkmale mit den Beschreibungen der Klassen in der Datenbasis des Systems.

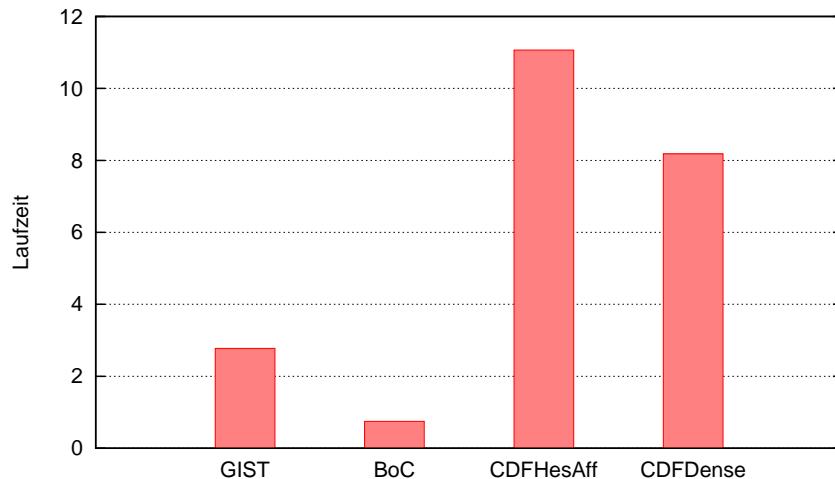


Bild 6.12: Vergleich der eingesetzten Merkmale bzgl. deren Laufzeiten bei der Klassifikation.

Es ist ein deutlicher Unterschied zwischen GIST bzw. BoC und den BoW-basierten Merkmalen festzustellen. Dies ist dadurch begründbar, dass bei GIST bzw. BoC je Bild und Klasse jeweils ein einziger Vektorvergleich erfolgt. Im Gegensatz dazu werden bei den klassenspezifischen Vokabularen je Klasse alle 20 visuellen Wörter (also 20 SIFT-Deskriptoren mit je 128 Dimensionen) mit allen interessanten Punkten des gegebenen

Bildes verglichen. Da in einem Bild der Auflösung 128x128 Pixel und unter Verwendung des Hessian-Affine-Detektors im Durchschnitt 275 interessante Punkte (bzw. bei densem Sampling jedes 8. Pixels insg. 256 Punkte) durch SIFT-Deskriptoren beschrieben werden, müssen $20 \cdot 275 = 5500$ (bzw. $20 \cdot 256 = 5120$) Vergleiche je Bild und Klasse durchgeführt werden. Für die Vorfilterung eignen sich basierend auf den Laufzeiten GIST und BoC am Besten.

6.5.2 Überlappung der nächsten Nachbarn

Um die Genauigkeit der Klassifikation bzw. der Annotation möglichst wenig zu beeinträchtigen, sollte das gewählte Merkmal allein die mit allen Merkmalen ermittelten besten N Klassen möglichst gut approximieren. Zur Feststellung, welches der verwendeten Merkmale die größten Überlappungen bzgl. der besten N Resultate aufweist, werden in Abbildung 6.13 die Merkmale bzgl. der Überlappung in Abhängigkeit von der prozentualen Einschränkung des Suchraumes mittels des in Abschnitt 5.2.2.2 vorgestellten Verfahrens untersucht. Die Werte stellen dabei den Mittelwert für alle 3251 Klassen unter Verwendung von 20 Testbildern je Klasse dar.

Betrachtet man die Approximation der besten N Resultate, setzt sich GIST deutlich von den anderen Merkmalen ab. Es ist anzumerken, dass durch eine gute Approximation der besten N Klassen noch nicht sichergestellt werden kann, dass auch die richtige Klasse sich in der Schnittmenge befindet. Um dies weitergehend zu untersuchen, werden im folgenden Abschnitt die relativen Verluste bzgl. der Genauigkeit der Klassifikation und damit verbunden die Qualitätseinbußen der Annotation untersucht.

6.5.3 Auswirkungen auf die Objekterkennung

Bei der Approximation der relevanten Klassenmenge für ein gegebenes Bild basierend auf einem Merkmal bzw. einer Untermenge aller im System verwendeten Merkmale muss ein möglichst optimales Gleichgewicht zwischen dem Maß der Einschränkung der Klassenmenge und den damit verbundenen Verlusten bei der Genauigkeit der Objekterkennung erzielt werden. In den Abschnitten 6.5.1 und 6.5.2 wurde GIST als das am besten geeignete Merkmal für die Vorfilterung der Klassenmenge identifiziert. Abbildung 6.14 veranschaulicht die relativen Einbußen, Abbildung 6.15 zeigt die durchschnittlichen Laufzeiten in Abhängigkeit der prozentualen Einschränkung der Klassenmenge durch eine

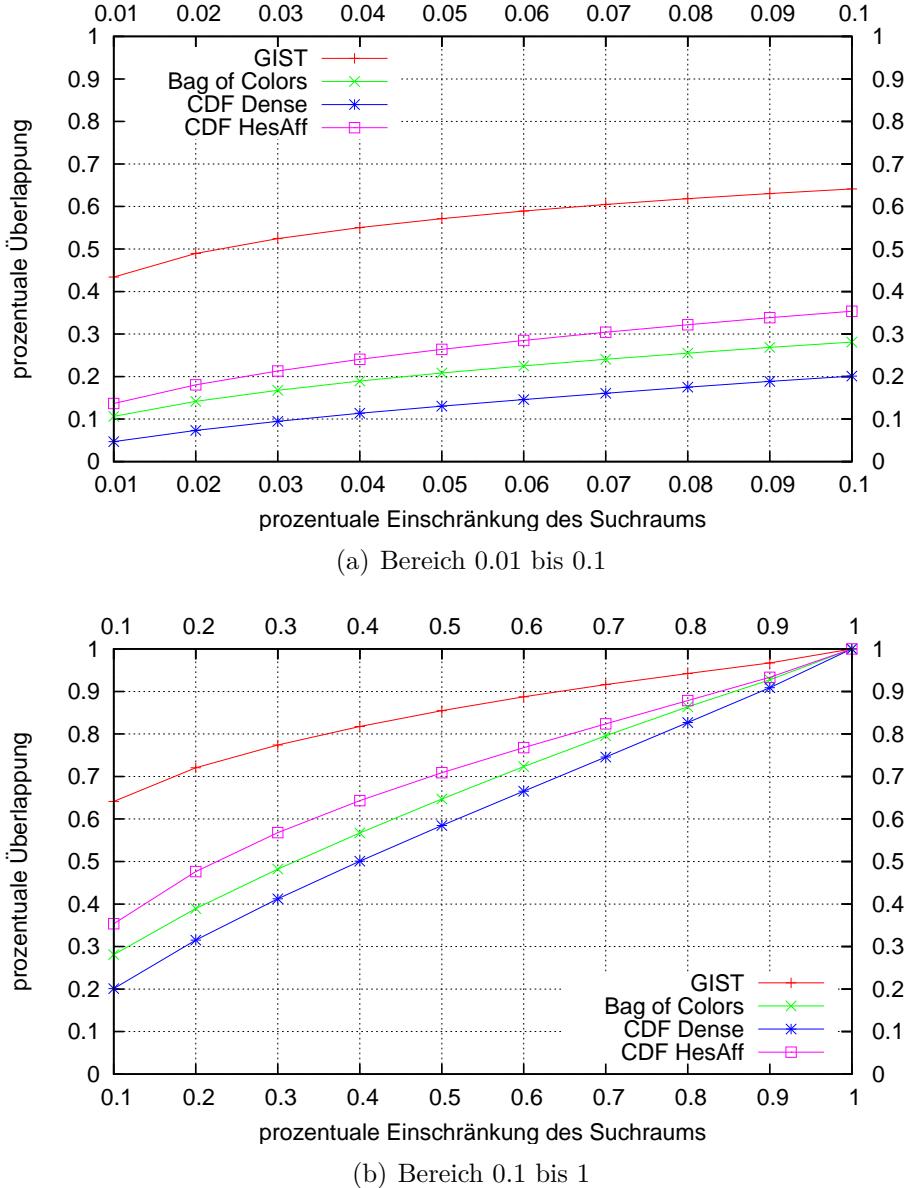


Bild 6.13: Vergleich der Merkmale bzgl. der Überlappung der nächsten Nachbarn mit den Klassifikationsergebnissen der kombinierten Anwendung aller Merkmale.

GIST-basierte Vorselektion im Vergleich zur Klassifikation ohne Vorfilterung. Die Werte stellen dabei den Mittelwert für 100 zufällig ausgewählte Testbilder und der Verwendung von 3 251 Klassen dar.

Die Laufzeit ohne Vorselektion fällt geringer aus als z. B. bei einer Einschränkung von 90%, da ohne Vorselektion die GIST-Vergleiche parallel zu den Vergleichen mit den

anderen Merkmalen ausgeführt werden und auch die Einschränkung der Klassenliste basierend auf dem Ergebnis der Vorfilterung entfällt.

6.5.4 Zusammenfassung

Ausgehend von der durchschnittlichen Berechnungszeit in Abschnitt 6.5.1 und der Überlappung der nächsten Nachbarn in Abschnitt 6.5.2 wurde für die Einschränkung des Suchraums der relevanten Klassen das GIST-Merkmal als am besten geeignet befunden. Mit der Einschränkung wird auch eine Ungenauigkeit eingeführt, welche zu einer Verschlechterung der Objekterkennung führt. In Abschnitt 6.5.3 wurde der Zusammenhang zwischen dem Maß der Einschränkung und die MAP evaluiert. Anhand dieser Untersuchung ist es möglich abzuwagen zwischen dem Gewinn an Zeit und dem Verlust an Genauigkeit. Innerhalb des Pixtract-Frameworks wird eine Einschränkung des Suchraums mittels der Vorfilterung auf 10% verwendet.

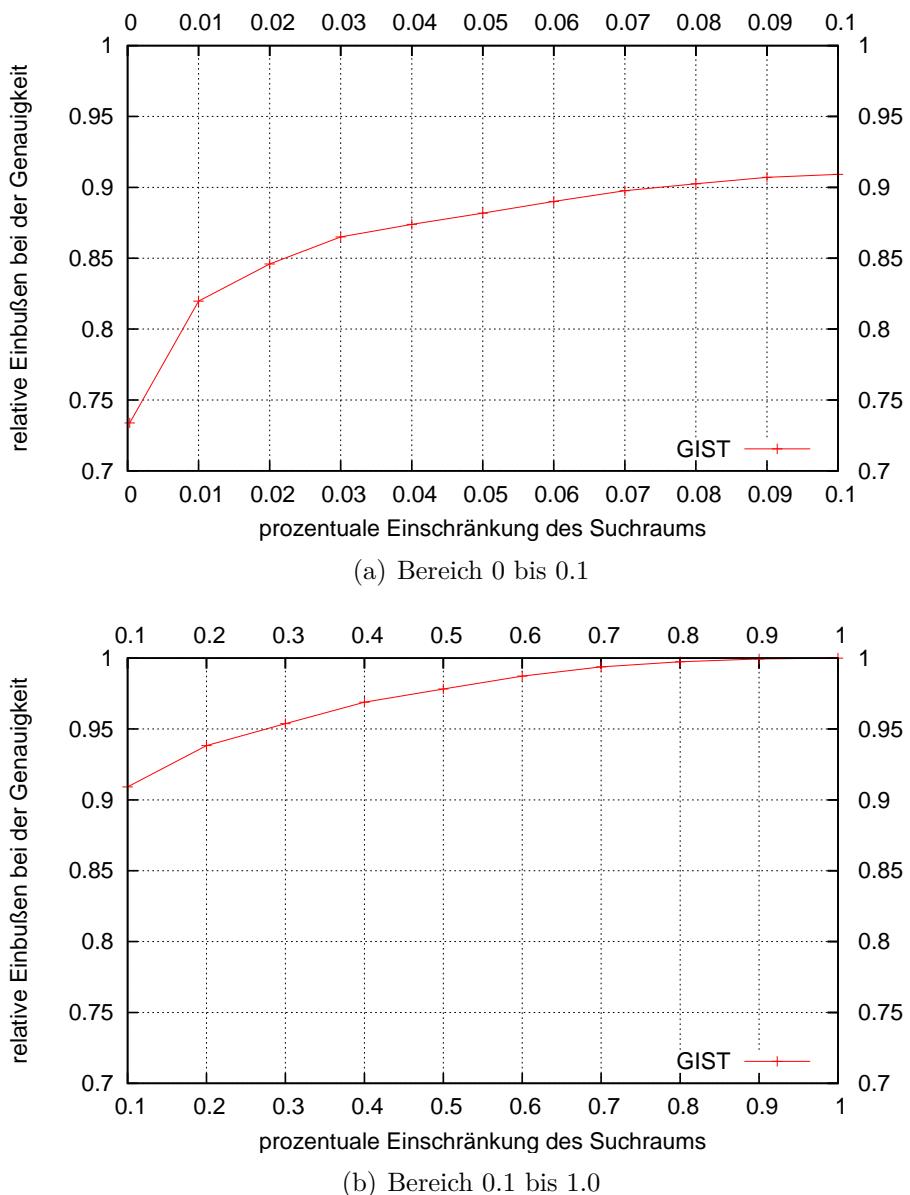
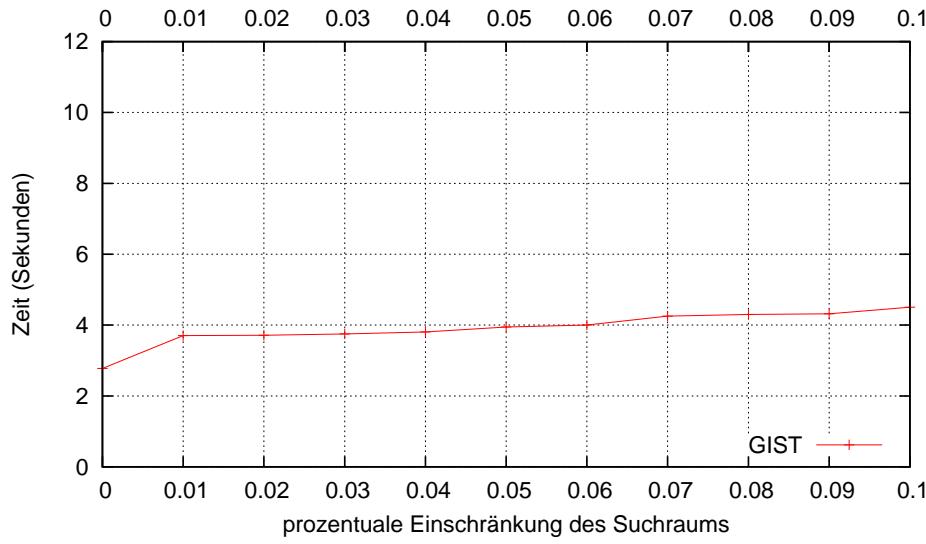
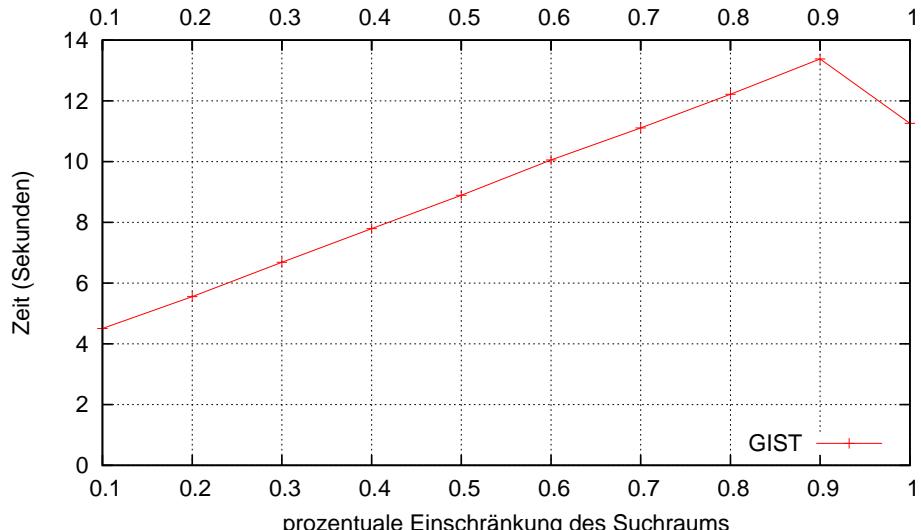


Bild 6.14: Beeinflussung der MAP bei der Verwendung einer Vorselektion basierend auf dem GIST-Merkmal in Abhängigkeit der prozentualen Einschränkung des Suchraumes.



(a) Bereich 0 bis 0.1



(b) Bereich 0.1 bis 1.0

Bild 6.15: Beeinflussung der Laufzeit bei der Verwendung einer Vorselektion basierend auf dem GIST-Merkmal in Abhängigkeit der prozentualen Einschränkung des Suchraumes.

6.6 Zusammenfassung

In diesem Kapitel wurde das erweiterbare Verfahren zur Objekterkennung in Bildern aus Kapitel 5 bzgl. der Genauigkeit der Erkennung, Skalierbarkeit und Ausführungszeit bei der Annotation optimiert. Beim kombinierten Einsatz aller Merkmale für 3 251 Klassen wurde eine ähnliche MAP erreicht wie in [DBLFF10], was dem Stand der Technik für mehrere tausend Klassen entspricht.

Um einen Überblick zu erhalten, wie das vorgestellte erweiterbare Verfahren bei verschiedenen Klassengruppen abschneidet, wurde die MAP für die meist vertretenen Klassengruppen des eingesetzten ImageNet-Datensatzes ermittelt. Die Mittelwerte der MAP für die hierarchisch aufgespaltenen Klassengruppen sind in Abbildung 6.16 dargestellt. Dabei wurden zur Übersichtlichkeit Gruppen mit weniger als 9 Unterklassen weggelassen.

Für die Berechnung des hierarchischen Fehlers mittels WordNet wurde das knotenbasierte Ähnlichkeitsmaß des ILSVRC-2011-Wettbewerbs aus Gleichung 4.23 eingesetzt. Die gemittelten hierarchischen Fehler für die meist vertretenen Klassengruppen des ImageNet-Datensatzes sind in Abbildung 6.17 dargestellt.

Aus Abbildung 6.16 lässt sich bzgl. der MAP-Werte ablesen, dass Tiere (*animal*) deutlich schlechter erkannt werden als vom Menschen erschaffene Objekte (*artifact*). Diese Schwierigkeit der Erkennung lässt sich auf die verschiedenen Stellungen, in denen Tiere auftreten können, zurückführen, was auch in Abschnitt 6.2.4 diskutiert wurde. Bezogen auf den hierarchischen Fehler in Abbildung 6.17 schneiden sowohl Lebewesen (*living thing*) als auch vom Menschen erschaffene Objekte (*artifact*) ähnlich gut ab. Das bedeutet, dass die korrekten und erkannten Konzepte innerhalb von WordNet ähnlich nah (bzw. weit) voneinander entfernt liegen.

Im folgenden Kapitel wird das Pixtract-Framework vorgestellt, welches Anwendern und Programmen den Zugriff auf das erweiterbare Objekterkennungssystem ermöglicht.

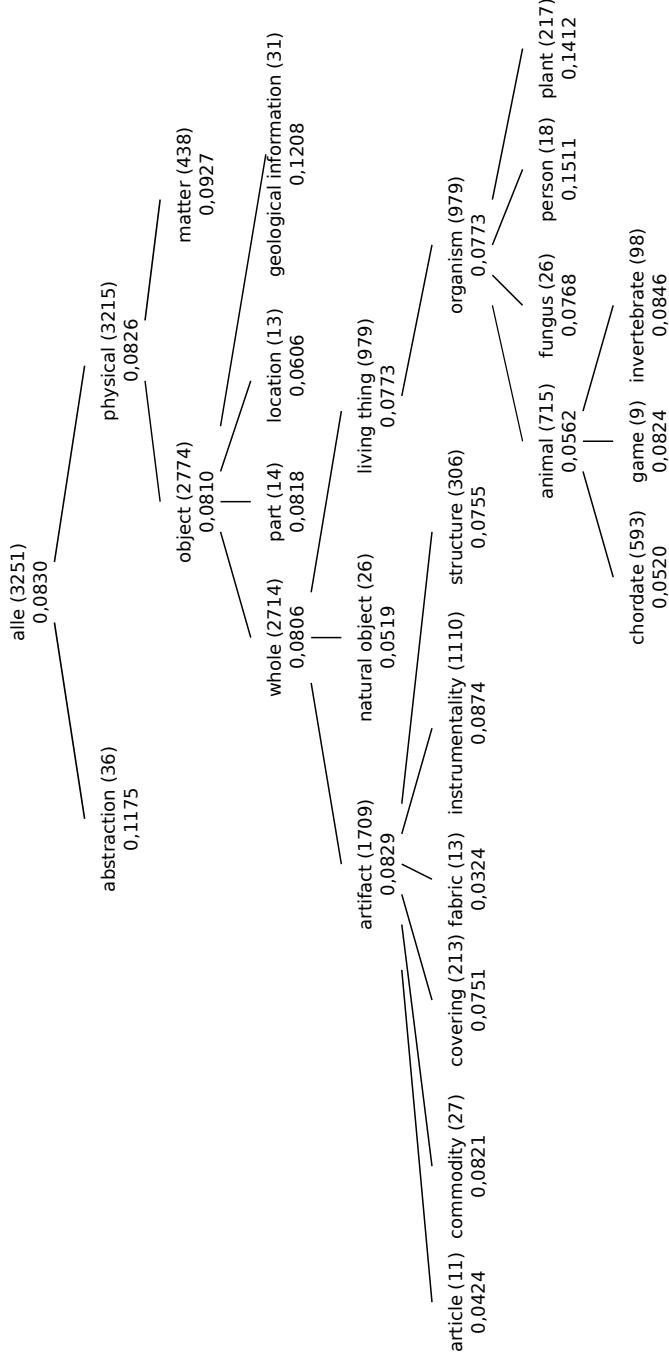


Bild 6.16: MAP für die Klassen der obersten Ebenen der Wordnet Hierarchie. In den Klammern ist die Anzahl der zugehörigen untergeordneten Klassen aus dem verwendeten ImageNet Datensatz angegeben. Der entsprechende MAP-Wert ist unter der jeweiligen Klassenbezeichnung aufgeführt.

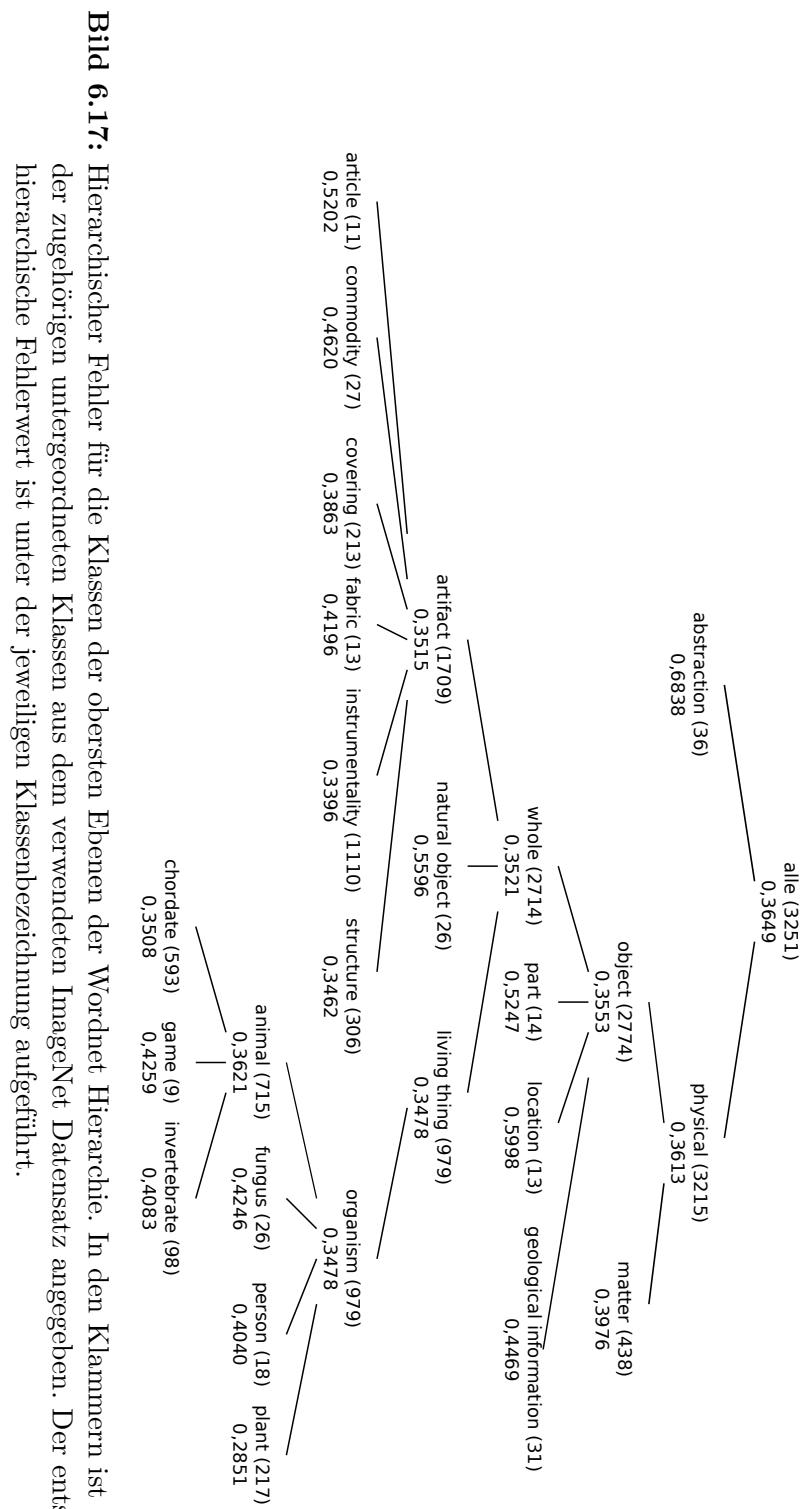


Bild 6.17: Hierarchischer Fehler für die Klassen der obersten Ebenen der Wordnet Hierarchie. In den Klammern ist die Anzahl der zugehörigen untergeordneten Klassen aus dem verwendeten ImageNet Datensatz angegeben. Der entsprechende hierarchische Fehlerwert ist unter der jeweiligen Klassenbezeichnung aufgeführt.

KAPITEL 7

ARCHITEKTUR DES PIXTRACT-FRAMEWORKS

Für die erweiterbare Annotation von Bildern wurde das Pixtract-Framework entworfen und implementiert. Die wesentliche Eigenschaft dieses Frameworks ist die Entkopplung der merkmal- und textbasierten Suche. Der Grund für diese Abgrenzung liegt zum einen darin, dass die Annotation der Bilder (also die merkmalbasierte Suche) länger dauern darf als später die textbasierte Suche auf den annotierten Bildern. Andererseits können bei der textbasierten Suche auch bereits etablierte Verfahren der Textindizierung und -Suche ohne jegliche Anpassungen oder Einschränkungen eingesetzt werden.

In den nachfolgenden Abschnitten werden die einzelnen Komponenten und Schnittstellen des Pixtract-Frameworks näher vorgestellt.

7.1 Grobarchitektur von Pixtract

Das Pixtract-Framework lässt sich in zwei wesentliche Komponenten aufteilen: die Lernkomponente zum Hinzufügen von neuen Klassen sowie die Annotationskomponente zur Annotation von Bildern basierend auf den Pixtract bekannten Klassen. Je nach Verwendung der ermittelten Annotation für die Bilder kann Pixtract um einen textba-

sierten Suchdienst erweitert werden. Der grobe Aufbau von Pixtract ist in Abbildung 7.1 dargestellt (vgl. Abbildung 1.2).

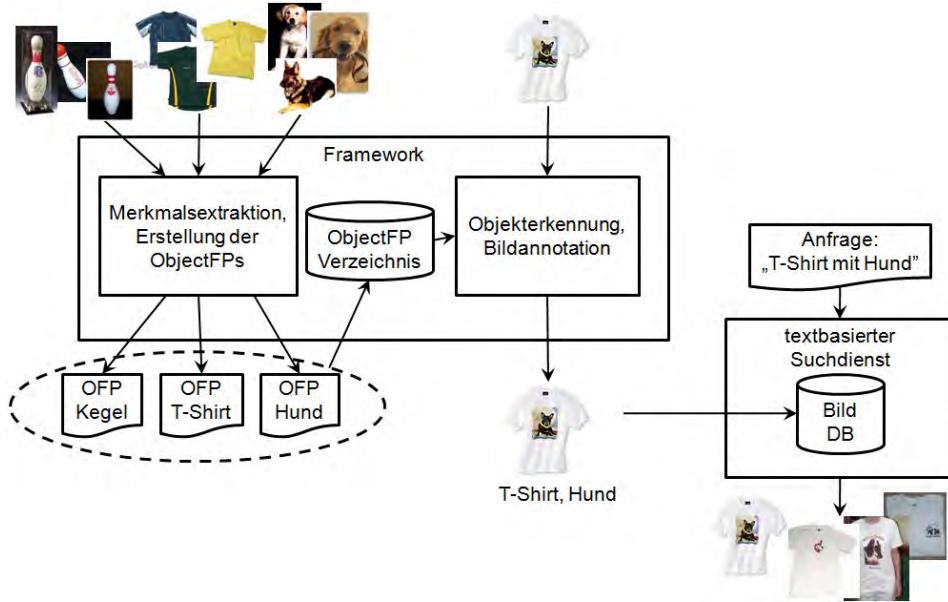


Bild 7.1: Grobarchitektur des Pixtract-Frameworks.

7.1.1 Grundkonzepte

Für das Erlernen von neuen Klassen und zur Annotation von Bildern ist eine einheitliche Menge von Konzepten (Begriffen, Kategorien) $C_k \in C$ einer Wissensbasis nötig. Erlernte Klassen $\Omega_\kappa \in \Omega$ werden Konzepten C_k zugeordnet und in neuen Bildern erkannte Objekte (bzw. Muster $f(\mathbf{x})$) werden schließlich als Instanzen I_j der zugehörigen Konzepte C_k als Beschreibung \mathcal{B} annotiert ($\mathcal{B} = \langle I_j(C_k) \rangle$, vgl. Gleichung 2.2.2).

Als Grundlagen für Konzepte C können Wissensbasen, wie z. B. Konzept-Thesauri oder Ontologien herangezogen werden. Einer der populärsten, größten und in der Objekterkennung vielfältig eingesetzten Konzeptsammlungen ist WordNet. Die Version 3.0 von WordNet wurde in Pixtract integriert und bildet die Grundmenge an Konzepten für die Unterscheidung von Klassen sowohl beim Hinzufügen von neuen Objekten als auch bei der Annotation von Bildern.

7.1.2 Lernkomponente

Pixtract ist ein erweiterbares System zur automatischen Annotation von Bildern. Dadurch erlaubt es das Hinzufügen von neuen Klassen zur Laufzeit und somit die dynamische Ergänzung des Klassenbestands, der zur Annotation von Bildern dient. Die Lernkomponente ermöglicht einer eingeschränkten Gruppe von Benutzern das Hochladen von Bildern zu einer neuen Klasse. Dazu müssen die Bilder von den Benutzern bereits bzgl. des abgebildeten und zu erlernenden Objekts vorselektiert sein. Weiterhin sollten die Bilder das Objekt möglichst vollständig, ohne Licht- oder Schatteneffekte und ohne Abdeckungen zeigen. Details und Begründungen für diese Anforderungen bzgl. der Trainingsbilder werden in Abschnitt 7.4 erläutert.

Nach dem Hochladen der Bilder für eine neue Klasse werden die Merkmale extrahiert und eine Beschreibung des Objekts (Objekt-Fingerabdruck, Object Fingerprint (ObjectFP)) erstellt. Diese ObjectFPs müssen einem Konzept aus WordNet eindeutig zugeordnet werden. Für das Auffinden des passenden Konzepts wird eine entsprechende Suchfunktionalität bereitgestellt.

Nach der Zuordnung des ObjectFPs zum gewählten Konzept wird die neue Klasse in das Objektverzeichnis von Pixtract aufgenommen und kann im Anschluss für die Annotation von Bildern verwendet werden.

7.1.3 Annotationskomponente

Die Annotationskomponente von Pixtract ermöglicht die Beschreibung von Bildern basierend auf den erkannten Objekten und der zugeordneten Konzepte. Dabei können bei der Annotation nur die Objekte erkannt werden, welche sich zum gegebenen Zeitpunkt im Objektverzeichnis von Pixtract befinden.

Für die Annotation können ein oder mehrere Bilder gleichzeitig hochgeladen werden. Nach Abschluss der Datenübermittlung werden aus den Bildern Merkmale extrahiert und diese mit den ObjectFPs von Pixtract verglichen. Als Ergebnis dieser merkmalbasierten Suche werden die zu den relevantesten ObjectFPs zugeordneten Konzepte als Annotation der Bilder ausgegeben. Die Annotation kann sowohl als einfacher Text direkt am Bildschirm betrachtet werden oder zur maschinellen Weiterverarbeitung in textbasierten Suchdiensten bzw. Bildverwaltungsprogrammen auch als XMP-Datei (siehe Abschnitt 3.3.1.3) heruntergeladen werden.

7.2 Umsetzung

Die Umsetzung der einzelnen Komponenten von Pixtract erfolgte in verschiedenen Teilprojekten.

Kernstück des Frameworks ist ein durch neue Merkmalsextraktoren einfach erweiterbares und konfigurierbares Programm namens *FeatExt*. Es dient zur Extraktion von Merkmalen aus Bildern sowie zur Generierung des Ergebnisses im XMP-Format. *FeatExt* wird in Abschnitt 7.2.1 kurz vorgestellt.

Für den Zugriff auf die Lern- und Annotationskomponente wurde eine webbasierte Benutzerschnittstelle aufbauend auf Linux, Apache, MySQL, PHP, Perl (LAMPP)¹ umgesetzt. Zusätzlich wurde auch eine REST-Schnittstelle implementiert, welche eine direkte Einbindung des Pixtract-Annotationsdienstes in externe Bildverwaltungsprogramme oder als Plugins in textbasierten Suchdiensten zur textuellen Beschreibung von Bildern ermöglicht. Beide Schnittstellen werden mit Beispielen zu Abläufen in Abschnitt 7.2.2 näher vorgestellt.

7.2.1 Konfigurierbare und erweiterbare Merkmalsextraktion

Zentrales Element des Pixtract-Frameworks ist das konfigurierbare und durch neue Merkmalsextraktoren einfach erweiterbare Programm *FeatExt*. Es wird sowohl von der Lernkomponente für die Berechnung der ObjectFPs aus einer Gruppe von Bildern als auch in der Annotationskomponente für die Extraktion von Merkmalen und die Erstellung der XMP-Dateien eingesetzt.

Die einfache Erweiterung durch neue Merkmalsextraktoren, Konverter oder andere Binärdateien bzw. Funktionen aus Bibliotheken wird durch einen Plugin-basierten Ansatz realisiert. Diese Plugins können in einer XML-Parameterdatei konfiguriert und ggf. auch verkettet werden, wie es z. B. beim BoW-Ansatz erforderlich ist. Ein Beispiel für eine Parameterdatei ist in Anhang E.1 zu sehen.

Die ermittelten Merkmale bzw. die Ausgaben der Plugins werden in einer XMP-Datei gesammelt. Dabei erhält jedes verwendete Plugin jeweils seinen eigenen Namensraum, in den es seine Daten ausgeben kann. Ein Beispiel für eine XMP-Ausgabedatei wird in Anhang E.2 gezeigt.

¹ <http://www.apachefriends.org/en/xampp-linux.html>

Nähere Informationen zum Entwurf, zur Implementierung sowie zur Bedienung von *FeatExt* können aus [Gal09] entnommen werden.

7.2.2 Schnittstellen

Pixtract bietet sowohl eine webbasierte Benutzerschnittstelle als auch eine an REST angelehnte Schnittstelle für die direkte Verwendung in Programmen an. Die Webschnittstelle erlaubt den Zugriff auf die Lern- und Annotationskomponenten, während die Programmschnittstelle lediglich die Annotation der Bilder ermöglicht. In den nachfolgenden Abschnitten werden die einzelnen Schnittstellen jeweils kurz vorgestellt.

7.2.2.1 Webschnittstelle

Die Webschnittstelle bietet dem Benutzer eine direkte Möglichkeit

- zum Durchsuchen der bereits erlernten Klassen von Pixtract bzw. von sonstigen WordNet-Konzepten,
- zur Annotation von Bildern sowie
- zum Hinzufügen von neuen Klassen anhand von Trainingsbildern.

Das Durchsuchen der vorhandenen Klassen im Objektverzeichnis von Pixtract ist allen Anwendern frei zugänglich. Für die Annotation von Bildern ist ein einfaches Benutzerkonto nötig. Da die Qualität der Bildannotation stark von den verwendeten Trainingsbildern abhängt, ist die Erweiterung von Pixtract um neue Klassen nur durch einen Administrator möglich.

Nachfolgend werden die einzelnen Funktionalitäten der Webschnittstelle kurz anhand von Beispielen vorgestellt. Weitergehende Informationen zum Entwurf, zur Implementierung und zur Bedienung der Webschnittstelle sind in [Uhl10] zu finden.

Visualisierung

WordNet dient in Pixtract als Wissensbasis für die Zuordnung von erlernten Objektklassen $\Omega_\kappa \in \Omega$ zu Konzepten $C_k \in C$. Die Webschnittstelle von Pixtract bietet dementsprechend eine Funktion zur Suche nach und Visualisierung von Konzepten an. Bei der Darstellung des Suchergebnisses erscheinen bereits erlernte Kategorien mit einem Thumbnail. Für die Visualisierung der über- und untergeordneten Begriffe wurde die Darstellung als

7. Architektur des Pixtract-Frameworks

2-Wege-Treemap in Anlehnung an [Shn92] gewählt. Die Suche, das Suchergebnis sowie die 2-Wege-Treemap für den Begriff „tennis“ sind in Abbildung 7.2 dargestellt.



Bild 7.2: Suche und Visualisierung von WordNet Synsets und Klassen in der Pixtract-Webschnittstelle.

Annotation

Für registrierte Benutzer ermöglicht die Webschnittstelle von Pixtract die Annotation von einem oder mehreren Bildern. Hierzu müssen die zu annotierenden Bilder ausgewählt und an Pixtract übertragen werden. Im Anschluss an die Datenübermittlung werden die Merkmale durch *FeatExt* aus den Bildern extrahiert und basierend auf den ObjectFPs des aktuellen Objektverzeichnisses die Konzepte zu den Bildern annotiert.

Für das Hochladen der Bilder wurde der Uploader¹ der Yahoo! User Interface Library (YUI) verwendet, welches sich durch seine intuitive Bedienung, leichte Integrierbarkeit und parallelen Dateiupload auszeichnet. Die erstellte Annotation wird direkt auf dem Bildschirm ausgegeben und kann zusätzlich für die maschinelle Weiterverarbeitung z. B. in einem Bildverwaltungsprogramm als XMP-Datei heruntergeladen und nachbearbeitet werden. Der Vorgang für die Annotation eines Bildes mittels der Webschnittstelle ist in Abbildung 7.3 dargestellt.

Erweiterung um neue Klassen

Die Erweiterung von Pixtract um neue Klassen ist nur über die Webschnittstelle möglich. Um eine neue Klasse hinzufügen zu können, muss der Anwender als Administrator angemeldet sein und eine Menge von vorselektierten Trainingsbildern bereit halten. Anschließend kann er nach dem zu erlernenden Konzept suchen und gemäß der kurzen Definition aus WordNet das passende Konzept auswählen. Dieser Vorgang ist in Abbildung 7.4 dargestellt. Nachfolgend werden die Trainingsbilder ausgewählt, an Pixtract mittels dem YUI Uploader übertragen und durch *FeatExt* der ObjectFP ermittelt. Nachdem der ObjectFP in das Objektverzeichnis von Pixtract integriert wurde, erscheint eine entsprechende Erfolgsmeldung sowie ein Thumbnail für die hinzugefügte Kategorie. Die entsprechenden Schritte sind in Abbildung 7.5 veranschaulicht. Das erzeugte Thumbnail ist anschließend bei der Suche nach vorhandenen bzw. erlernten Klassen neben dem zugehörigen WordNet Synset zu sehen (vgl. Beispiel für Visualisierung in Abbildung 7.2).

¹ <http://developer.yahoo.com/yui/uploader/>

7. Architektur des Pixtract-Frameworks

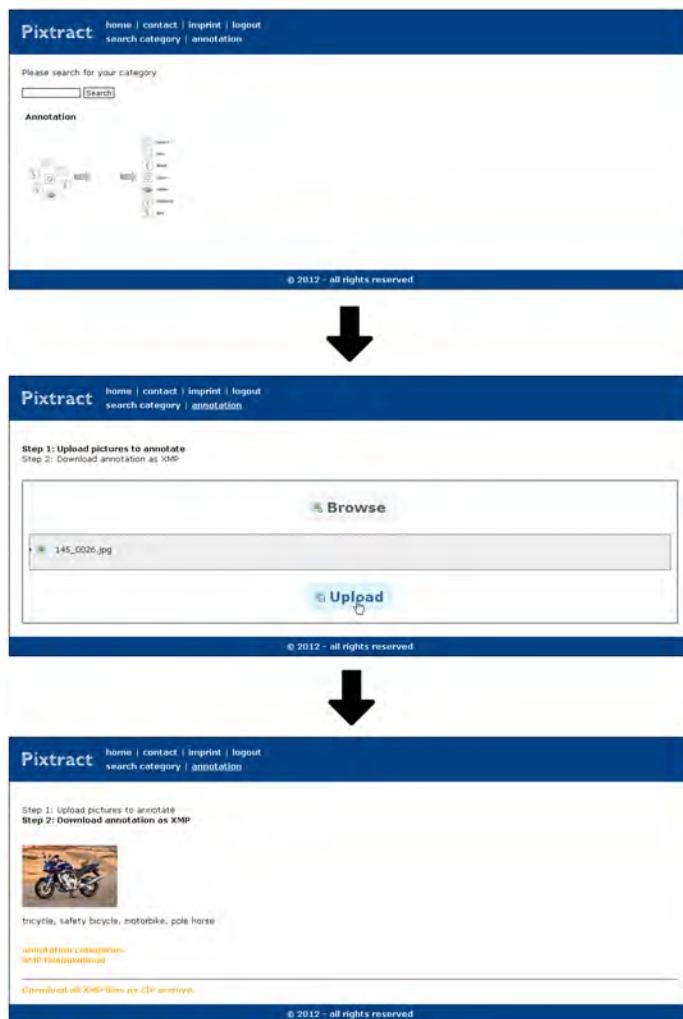


Bild 7.3: Ablauf der Annotation mittels der Pixtract-Webschnittstelle.

7.2.2.2 PicID-Schnittstelle

Um eine reibungslose Integration des Annotations- und Suchdienstes zu ermöglichen, wird eine PicID-Schnittstelle in Anlehnung an MusicID¹ bereitgestellt. Diese ermöglicht die Abwicklung der Annotation von Bildern über ein Plugin, welches in beliebige Bildverwaltungsprogramme oder textbasierte Suchdienste integriert werden kann. Der Benutzer kann somit direkt in seinem Bildverwaltungsprogramm die Annotation anstoßen, wobei die gesamte Kommunikation sowie XMP-Verarbeitung im Hintergrund vom entsprechenden Plugin durchgeführt wird. Als Ergebnis liegt nach der Transaktion die Annotation

¹ <http://www.gracenote.com>



Bild 7.4: Ablauf der Hinzufügung einer neuen Klasse mittels der Pixtract-Webschnittstelle
– Teil 1.

zum Bild vor, ähnlich wie bei MusicID der Titel, das Album und der Interpret zu einem Musikstück.

Die PicID-Schnittstelle wurde als eine einfache Client-Server-Anwendung implementiert. Auf der Seite von Pixtract nimmt ein Server HTTP-Anfragen von Clients entgegen. Eine

7. Architektur des Pixtract-Frameworks

Pixtract home | contact | imprint | logout
search category | annotation | learn category | user administration

Learn a new category
Step 1: Search Category
Step 2: Choose the right category
Step 3: Upload pictures for category
Step 4: New category learned - finished

Please upload your pictures for car - a wheeled vehicle adapted to the rails of railroad

1470.jpg
0904.jpg
1840.jpg
0405.jpg
0681.jpg
1109.jpg
1227.jpg
2396.jpg

© 2012 - all rights reserved

Pixtract home | contact | imprint | logout
search category | annotation | learn category | user administration

Learn a new category
Step 1: Search Category
Step 2: Choose the right category
Step 3: Upload pictures for category
Step 4: New category learned - finished

New category learned
car

Thank you for supporting Pixtract!

learn another category

© 2012 - all rights reserved

Pixtract home | contact | imprint | logout
search category | annotation | learn category | user administration

Please search for your category
car

car a motor vehicle with four wheels; usually propelled by an internal combustion engine visualisieren	car wheel a vehicle adapted to the rails of railroad visualisieren	car compartment the compartment that is suspended from an airship and that carries personnel and the power plant visualisieren	car where passengers ride up and down visualisieren
car battery a lead-acid storage battery in a motor vehicle; usually a 12-volt battery of six cells; the heart of the car's electrical system visualisieren	car bomb a bomb placed in a car and wired to explode when the ignition is started or by remote control or by a timing device visualisieren	car boot sale an outdoor sale at which people sell things from the trunk of their car visualisieren	car care keeping a car in good working order visualisieren
car company a company that makes and sells automobiles visualisieren	car dealer a firm that sells and buys cars visualisieren	car door the door of a car visualisieren	car factory a factory where automobiles are manufactured visualisieren
car horn a device on an automobile for making a warning noise visualisieren	car insurance insurance against loss due to theft or traffic accidents visualisieren	car loan a personal loan to purchase an automobile visualisieren	car maker a business engaged in the manufacture of automobiles visualisieren
car manufacturer a business engaged in the manufacture of automobiles visualisieren	car mirror a mirror that the driver of a car can use visualisieren	car park a lot where cars are parked visualisieren	car part a component of an automobile visualisieren
car pool a small group of car drivers who arrange to take turns driving and the others are passengers visualisieren	car port garage for one or two cars consisting of a roof supported on poles visualisieren	car race a race between (usually high-performance) automobiles visualisieren	car racing the sport of racing automobiles visualisieren
car rental a rented car visualisieren	car seat a seat in a car visualisieren	car sickness motion sickness experienced while traveling in a car visualisieren	

<http://localhost/index.php?page=pageLearnCategory.php?itemID=18255942>

Bild 7.5: Ablauf der Hinzufügung einer neuen Klasse mittels der Pixtract-Webschnittstelle – Teil 2.

Anfrage enthält als Daten ein Bild. Für die Annotation von mehreren Bildern wird je Bild eine separate Anfrage vom Client an den Server geschickt. Die erstellte Annotation wird als XMP in eine Antwortnachricht verpackt und an den Client zurückgeschickt. Da Bilddateien den gleichen Namen besitzen können – vor allem, wenn sie von der Kamera automatisch vergeben wurden –, wird zur Identifikation eines Bildes dessen Hash-Wert verwendet. Der Hash-Wert wird mittels der SHA-2 (SHA-256) Hashfunktion von ImageMagick¹ [Sti06] berechnet.

Details zum Entwurf und zur Implementierung der PicID-Schnittstelle von Pixtract für externe Bildverwaltungsprogramme und textbasierte Suchdienste sind in [Pro11] zu finden.

Einbindung in externe Programme

Pixtract ermöglicht sowohl für Bildverwaltungsprogramme als auch für textbasierte Suchdienste die Annotation von Bildern. Somit kann z. B. Lucene² [MHG10] erweitert werden und neben Text auch in Bildern – basierend auf deren Annotation – suchen. Bei Bildverwaltungsprogrammen ist es von der jeweiligen Anwendung abhängig, inwieweit die Annotation weiterverarbeitet wird. Im einfachsten Fall wird lediglich eine XMP-Datei im oder neben dem Bild abgespeichert. Bei umfangreicheren Bildverwaltungsanwendungen können die XMP-Dateien auch weiterverarbeitet und ggf. in einen Index aufgenommen werden. Diese Art der Verarbeitung kommt dem Szenario eines textbasierten Suchdienstes gleich.

Als Beispiel für ein Bildverwaltungsprogramm wurde gThumb³ verwendet, da es leicht durch neue Funktionalitäten erweitert werden kann. Bei der nahtlosen Einbettung in gThumb können mehrere Bilder ausgewählt und anschließend durch das Menü die Annotationen in Form von XMP-Dateien angefordert werden. Dieser Vorgang ist in Abbildung 7.6 dargestellt. Dabei wird im Hintergrund der Client mit dem Bild bzw. den Bildern als Parameter aufgerufen. Dieser wickelt die Kommunikation mit dem Server ab und stellt im Anschluss die erhaltenen XMP-Dateien für gThumb bereit. In gThumb selber ist von dem gesamten Vorgang lediglich ein Fortschrittsbalken zu sehen. Da

1 <http://www.imagemagick.org>

2 <http://lucene.apache.org>

3 <http://gthumb.sourceforge.net>

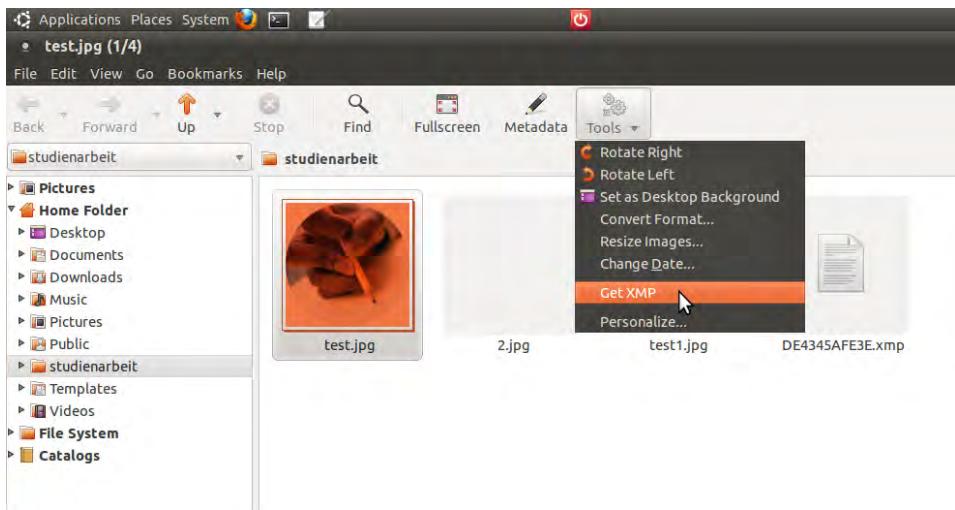


Bild 7.6: Annotation von Bildern eingebettet in die Bildverwaltungsanwendung gThumb.

gThumb keine Indizierung von Bildern basierend auf deren Metadaten anbietet, werden die XMP-Dateien einfach im selben Ordner wie das Bild bzw. die Bilder abgelegt.

Für das Szenario der Ankopplung an einen textbasierten Suchdienst wird Lucene als Beispiel verwendet. Lucene verarbeitet ausschließlich Text. Um auch die Verarbeitung von Bildern zu ermöglichen, wird Pixtract als Annotationsdienst eingesetzt. Für einen reibungslosen Ablauf beim Laden von Bildern in den Index wird Lucene um einen Plugin erweitert. Dieser nimmt die Bilder entgegen, wickelt die Kommunikation mit Pixtract ab, empfängt die XMP-Dateien, analysiert diese und stellt die Annotationen zu den Bildern für Lucene bereit. Die Annotationen werden inklusive der Bilder in den Lucene-Index aufgenommen, wobei die Bilder – ähnlich wie auch BLOBs in Datenbanken – in einem festgelegten Ordner gespeichert werden und im Index nur der Pfad des Bildes vermerkt wird. Zuletzt wird die Suche noch dementsprechend angepasst, dass zusätzlich zu den Stichwörtern auch die zugehörigen Bilder als Ergebnis zurückgeliefert werden.

7.3 Parallelisierung

Um neue Klassen dem System schneller hinzuzufügen und die Zeit für die Annotation von neuen Bildern zu verkürzen, können zusätzlich zu den Methoden in Abschnitt 5.2 Parallelisierungstechniken eingesetzt werden. Potenziale für die Parallelisierung ergeben sich bei der Extraktion der Merkmale aus Bildern, der Erstellung der Klassenbeschrei-

bungen sowie den Distanzberechnungen für die NN-Suche. In diesem Abschnitt wird ein kurzer Überblick zu gängigen Parallelisierungsmöglichkeiten gegeben.

Die direkte Möglichkeit für die Entwicklung paralleler Programme ist die Verwendung der POSIX Thread API, das oft auch als *pthreads* bezeichnet wird. Durch den Einsatz von leichtgewichtigen Threads kann zwar die Rechenleistung von mehreren Kernen bzw. Prozessoren ausgenutzt werden, jedoch sind erhebliche Anpassungen im Programmcode dafür nötig. Außerdem beschränkt sich die parallele Ausführung nur auf lokale Rechner oder Server. Abhilfe für diese Nachteile schaffen OpenMP und MPI, die sich beide quasi als Standard für die Entwicklung von parallelen Programmen etabliert haben.

Open Multi Processing (OpenMP) [HL09] ermöglicht durch seine Programmierschnittstelle die einfache Integration von Parallelität in C/C++, Fortran oder auch Java Programmen. Die entsprechenden Bereiche im Quellcode werden mit `#pragma` Direktiven versehen, welche vom Compiler so übersetzt werden, dass die enthaltenen Anweisungen in mehreren Threads parallel ausgeführt werden. Vorteil von OpenMP ist, dass der ursprüngliche Programmcode unberührt bleibt, d. h. sofern die Option für OpenMP dem Compiler nicht übergeben wird, werden die entsprechenden Direktiven ignoriert und der Programmcode kann trotzdem kompiliert werden. Zusätzlich unterstützt OpenMP die meisten gängigen Betriebssysteme. Wesentlicher Nachteil ist jedoch, dass OpenMP nur für Shared-Memory-Architekturen bestimmt ist. Durch Erweiterungen von OpenMP (z. B. in Verbindung mit MPI) ist jedoch auch der Einsatz in verteilten Systemen möglich, wo dann mehrere Cluster mit Shared Memory untereinander verbunden werden.

Message Passing Interface (MPI)¹ [GLDS96] ist eine programmiersprachenunabhängige API-Spezifikation zur nachrichtenbasierten Kommunikation zwischen Prozessen. Unter Zuhilfenahme von MPI können parallele Programme erstellt werden, wobei MPI nichtlokale Speicherzugriffe stets zu vermeiden versucht. Wegen der effizienten hardware-optimierten Implementierungen und der hohen Portabilität ist MPI quasi zum Standard für parallele Anwendungen auf Systemen mit verteiltem Hauptspeicher geworden. Wesentlicher Nachteil von MPI ist zur Zeit noch die eingeschränkte Fehlertoleranz z. B. beim Ausfall von Rechnern während der Laufzeit.

Nachteil der vorgestellten Methoden ist, dass viele Schwierigkeiten der Parallelisierung, Fehlertoleranz, Verteilung der Daten sowie Lastverteilung direkt beim Erstellen der Programme angegangen werden müssen. Aus diesem Grund wurde in [DG04, DG08]

¹ <http://www mpi-forum.org>

MapReduce als Programmiermodell vorgeschlagen, welches an den Funktionen *map* und *reduce* aus Lisp bzw. ähnlichen Programmiersprachen angelehnt ist. MapReduce ermöglicht die einfache Erstellung von Programmen zur Bearbeitung von großen Datensätzen auf beliebig großen Clustern. Der Ansatz basiert auf Schlüssel-Wert-Paaren, welche von der *map*-Funktion aus den Eingabedaten generiert werden. Paare mit dem selben Schlüssel werden vom Framework gruppiert und der *reduce*-Funktion übergeben, welche basierend auf den Werten weitere Berechnungen durchführt. Typischerweise werden von den *reduce*-Funktionen entweder keine oder nur eine Ausgabe erzeugt.

Welcher Rechner innerhalb des Clusters welche *map*- bzw. *reduce*-Funktionen ausführen soll, wird vom Master-Knoten entschieden. Das Framework übernimmt dabei die Lastverteilung, das Aufsetzen von Ersatztasks und die Behandlung von Ausfällen. Die Verteilung der Daten wird durch die Verwendung eines verteilten Dateisystems (z. B. Google File System (GFS) [GGL03]) oder einem Key-Value Storage (z. B. Bigtable [CDG⁺06, CDG⁺08]) gelöst. Diese Aspekte bleiben dem Programmierer komplett verborgen und werden allgemein im System optimiert.

Wegen der genannten Eigenschaften hat sich MapReduce in der Praxis sehr gut bewährt. Zum Anfang wurde es bei Google zur Indizierung von Webseiten eingesetzt, später wurde die Konfiguration der Server und das Aufsetzen von MapReduce-Programmen durch verschiedene Cloud-Dienstleister vereinfacht, wie z. B. bei Amazon Elastic MapReduce¹. Das MapReduce-Framework ist für zahlreiche Programmiersprachen verfügbar. Die bekannteste Implementierung unter ihnen ist das in Java umgesetzte Apache-Projekt Hadoop². Zur Dateiverwaltung kann das Hadoop Distributed File System (HDFS) (welches GFS nachempfunden wurde) oder HBase (welches in Anlehnung an Bigtable umgesetzt wurde) verwendet werden. Eine detaillierte Einführung in Hadoop ist in [Whi10] zu finden.

Die Schwächen bzgl. der Ausführungszeit in Hadoop wurden in letzter Zeit in Projekten wie z. B. Hadoop++ [DQRJ⁺10] oder Sector-Sphere [GG09] verbessert. Auch die Verbindung vom maschinellen Lernen oder Data Mining mit Hadoop wurde in Projekten wie z. B. Apache Mahout³ [CKL⁺06], Ricardo (R und Hadoop) [DSB⁺10] oder Radoop (eine Verbindung von Mahout, Hadoop und Rapidminer) [PMHGP11] ermöglicht.

1 <http://aws.amazon.com/elasticmapreduce/>

2 <http://hadoop.apache.org/>

3 <http://lucene.apache.org/mahout/>

In der vorliegenden Arbeit wurde die Parallelisierung der Merkmalsextraktion und Distanzberechnungen in Anlehnung an MapReduce durch eine vereinfachte Variante gelöst. Als verteiltes Dateisystem wird NTFS eingesetzt, die einzelnen Prozesse werden mit SSH auf die verfügbaren Rechner verteilt. Die parallele Ausführung konnte somit ohne jeglichen Eingriff in den Programmcode oder Behinderungen durch Framework-bedingte Verwaltungsstrukturen realisiert werden.

7.4 Säuberung der Stichprobe

Um die bestmögliche Erkennungsrate zu erreichen, ist es wichtig saubere Trainingsdaten zu verwenden. Im Pixtract-Framework stammt die Stichprobe von einem eingeschränkten Kreis von Anwendern des Systems. Die Bilder werden für eine gegebene Klasse bereits vom Benutzer vorselektiert und anschließend nach Pixtract hochgeladen. Dabei können einige Bilder doppelt vorkommen, eine zu kleine Größe aufweisen oder stark von den anderen hochgeladenen Bildern abweichen. Letztere können durch Abdeckungen, Ausschnitte, extreme Aufnahmewinkel und Vergrößerungen, (Tiefen-)Unschärfe sowie Beleuchtungs- und Schatteneffekte entstehen. Es kann jedoch auch ein simples Versehen der Grund für starke Abweichungen sein, wenn z. B. ein Bild einer anderen Klasse durch Zufall oder Unachtsamkeit des Anwenders in der falschen Menge landet. Um solche Bilder aus der Stichprobe auszuschließen, ist ein Filtermechanismus für die Eingabe nötig.

Ein ähnliches Problem der Filterung besteht beim initialen Laden einer Grundmenge von Klassen, welche zur Evaluation des Pixtract-Frameworks mit einer großen Anzahl von Klassen benötigt wird. Als Quelle für die in Klassen sortierten Bilder wurde ImageNet [DDS⁺09] herangezogen, da es zur Zeit der größte öffentlich verfügbare Datensatz ist mit mehr als 20 000 manuell eingeordneten Klassen. Ein weiterer Vorteil von ImageNet ist, dass die einzelnen Klassen Konzepte aus WordNet direkt zugeordnet sind und es somit in Pixtract, was ebenfalls WordNet als Wissensbasis verwendet, leicht zu integrieren ist. Von Interesse sind insbesondere diejenigen Klassen, bei denen die Objekte in den Bildern auch mit Rahmen markiert sind, da diese Teilbilder nur das Objekt selbst beinhalten.

Leider befinden sich auch unter den bereits vorsortierten Bildern von ImageNet einige Aufnahmen, die extreme Abdeckungen, Ausschnitte, Unschärfe oder Schatteneffekte aufweisen. Zur Erstellung einer sauberen Trainingsmenge basierend auf den Bildern von ImageNet ist somit ebenfalls ein Filtermechanismus nötig.

7.4.1 Filtermechanismus basierend auf dem GIST-Deskriptor

Die Anforderungen an einen Filter für die Säuberung der Trainingsbilder können in folgenden Punkten zusammengefasst werden:

1. Aussieben von Bildern mit zu geringer Größe, da diese Bilder nur wenig Information enthalten;
2. Aussieben von Duplikaten und beinahe Duplikaten, da diese Bilder keine zusätzliche Information mit sich bringen;
3. Ermittlung und Ausschluss von Ausreißern (Outlier), die stark von den restlichen Bildern abweichen.

Der Filtermechanismus ermittelt die besten (bzw. bei Umkehrung die schlechtesten) N Bilder für eine gegebene Stichprobe ω der Klasse Ω_κ mit der Größe $|\omega|$, wobei für eine wirkungsvolle Filterung die Bedingung $|\omega| > N$ gelten muss. Die einzelnen Schritte der Säuberung sind in Abbildung 7.7 dargestellt und werden nachfolgend im Detail beschrieben.

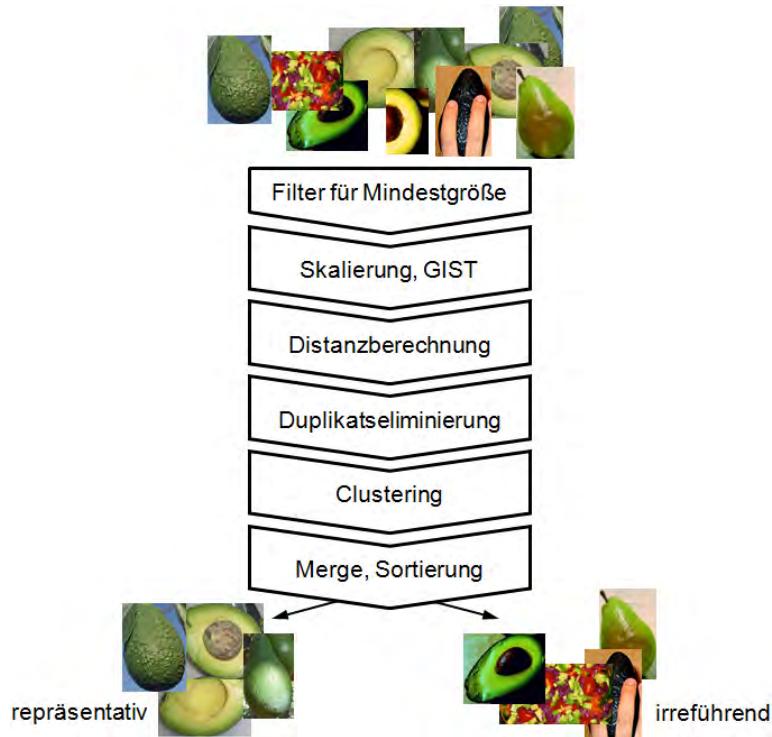


Bild 7.7: Ablauf des GIST-basierten Filters zur Eliminierung von Ausreißern und zur Bestimmung der besten Trainingsbilder am Beispiel der Klasse „avocado“.

Bildgröße

Für das Aussieben der kleinen Bilder wird unter Zuhilfenahme der ImageMagick-Schnittstelle die Höhe und die Breite eines gegebenen Bildes ermittelt. Sofern beide Seiten kleiner als 128 Pixel sind, wird das aktuelle Bild ausgeschlossen. Die Schranke wurde deswegen auf 128 Pixel festgesetzt, weil für die Extraktion von Merkmalen die Bilder zum Teil auf diese Seitenlänge herunterskaliert werden. Bilder, die entweder nur in der Höhe oder der Breite die Mindestgröße unterschreiten, werden beibehalten, da bei länglichen Objekten, wie z. B. Skateboard oder Baseballschläger, sonst kaum Bilder übrig bleiben würden.

Duplikatseliminierung

Für die Eliminierung von Duplikaten und beinahe Duplikaten wird der GIST-Deskriptor verwendet, da er sich bereits vielfach für die Erkennung von Duplikaten bewährt hat und nach Abschnitt 6.5.1 schnell berechnen bzw. vergleichen lässt. Da GIST auch mit kleinen Bildern gut funktioniert (wie z. B. in [TFW08] oder [DJS⁺09] gezeigt), werden die Bilder zuerst auf 128x128 Pixel herunterskaliert, wobei das ursprüngliche Seitenverhältnis ignoriert wird. Aus diesen verkleinerten Bildern werden im Anschluss GIST-Deskriptoren ähnlich wie in Abschnitt 6.2 mittels der Implementierung von [DJS⁺09] berechnet. Basierend auf den GIST-Deskriptoren wird mittels der euklidischen Distanz eine Distanzmatrix mit allen Bildern berechnet. Anschließend wird die Distanzmatrix auf Werte kleiner als eine vorgegebene Schranke ε untersucht. Distanzen, die diesen ε -Wert unterschreiten, werden als Duplikate bewertet und eines der beiden Bilder wird ausgeschlossen.

Ermittlung von Outliern

Für die Ermittlung der besten bzw. schlechtesten Bilder einer Klasse wird ein Clustering-basiertes Verfahren in Anlehnung an [LWZ⁺08] angewendet. Die Idee hinter diesem Ansatz ist, dass relevante und saubere Aufnahmen innerhalb einer Klasse häufiger vorkommen werden und somit größere Cluster bilden, während Ausreißer kleinere oder sogar Cluster mit einem einzigen Element erzeugen.

Wesentlich ist die Bestimmung der Anzahl k der Cluster, da die Cluster weder zu groß noch zu klein werden sollten. Der Parameter k ist dabei abhängig von der Anzahl der Bilder und deren Verteilung im gewählten Merkmalsraum. Da die Anzahl der Bilder variieren kann, wurde k auf einen prozentualen Anteil der Anzahl der Bilder festgesetzt.

Es wurden verschiedene Prozentwerte ausprobiert, wobei stets die resultierende Anzahl der Cluster und deren Population in Betracht gezogen wurden. Die besten Resultate entstanden bei einem Wert um 20%. Also z. B. bei $|\omega| = 200$ Bildern ergibt sich $k = 40$ als die Anzahl der Cluster. Dies entspricht einer durchschnittlichen Anzahl von 5 Bildern je Cluster, im Regelfall entstehen jedoch wenige große und viele kleine Cluster (z. T. auch nur mit einem einzigen Element). Da keine Annahme getroffen werden kann, wie stark eine Klasse mit Ausreißern übersät ist, wurde dieser Prozentwert für alle Klassen auf den gleichen Wert festgesetzt.

Für die Aufteilung der Bilder in k Cluster wird ein hierarchisches Verfahren angewendet, wobei die bereits berechnete Distanzmatrix basierend auf den GIST-Deskriptoren als Grundlage dient. Die Cluster werden im Anschluss absteigend nach ihrer Population sortiert und die größten Cluster solange verbunden, bis die Anzahl der Bilder einen vorgegebenen Wert N übersteigt. Danach wird der meanGIST-Vektor des neuen verbundenen Clusters ermittelt und die Distanzen zu allen Mitgliedern berechnet. Die Bilder werden aufsteigend nach ihrer Distanz geordnet und die ersten N als die besten Bilder ausgegeben.

Für die Ermittlung von Ausreißern können entweder alle Cluster mit jeweils nur einem Element ausgegeben oder die Umkehrung der oben vorgestellten Methode für die N besten Bilder durchgeführt werden. Dabei werden die kleinsten Cluster angefangen bei Clustern mit nur einem Element (unabhängig von deren Entfernung zueinander) solange verbunden, bis die Anzahl der Bilder insgesamt N übersteigt. Anschließend wird der meanGIST-Vektor des neuen verbundenen Clusters berechnet und die Distanzen zu allen Mitgliedern ermittelt. Ausgegeben werden diejenigen Bilder, welche die größte Distanz zum Durchschnitt haben.

Beispielergebnisse des vorgestellten GIST-Filters für einige ausgewählte ImageNet-Klassen sind im Anhang F dargestellt.

7.4.2 Auswirkungen auf die Objekterkennung

Es wird erwartet, dass eine saubere Trainingsmenge zu besseren Ergebnissen bei der Objekterkennung führt. Zur Bewertung der Auswirkungen wurden die mit der in Abschnitt 7.4.1 vorgestellten Methode als beste Darstellungen einer Klasse ermittelten Bilder mit zufällig ausgewählten Trainingsbildern verglichen. Als Basis dienten 3 251 mit Bounding Boxes annotierte Klassen aus ImageNet. Zum Training wurden 20 gefilterte

oder zufällig ausgewählte Bilder aus jeder Klasse herangezogen. Zur Evaluation dienten je Klasse 20 Testbilder, welche ebenfalls eine Mindesthöhe oder Mindestbreite von 128 Pixel aufweisen mussten und von denen die Duplikate eliminiert wurden. Da Trainings- und Testbilder disjunkt sein und eine Mindestgröße haben müssen, sind einige Klassen wegen zu wenigen entsprechenden Bildern weggefallen. Für die Evaluation blieben letztendlich 3 069 Klassen übrig.

Abbildung 7.8 veranschaulicht für verschiedene ausgewählte Merkmale die Auswirkungen auf die Erkennung von Objekten bei Verwendung der besten Trainingsbilder. Für jedes Merkmal werden drei Szenarien verglichen, wobei je Klasse immer 20 Trainingsbilder ausgewählt wurden:

- zufällige Auswahl der Trainingsbilder ohne vorherige Filterung,
- Forderung einer Mindestseitenlänge von 128 Pixel und anschließend zufällige Auswahl der Trainingsbilder,
- Säuberung der Bildermenge nach dem in Abschnitt 7.4.1 beschriebenen GIST-basierten Verfahren und Auswahl der besten Bilder als Trainingsbilder.

Für jede der drei Szenarien wurden 20 Testbilder zufällig ausgewählt, wobei für die letzten zwei Szenarien die Größe der Testbilder ebenfalls auf eine Mindestlänge von 128 Pixel beschränkt wurde. Da die Testbilder disjunkt zu den Trainingsbildern sein müssen, ist die Menge der Testbilder für die jeweiligen Szenarien unterschiedlich.

In Abbildung 7.9 wird der Einfluss der Sauberkeit der Trainingsbilder auf die Objekterkennung veranschaulicht. Klassen, deren Deskriptoren aus anhand des in Abschnitt 7.4 vorgestellten Verfahrens gefilterten Trainingsbildern berechnet wurden, schneiden durchweg besser ab als die mit zufällig ausgewählten Trainingsbildern.

Im Vergleich mit der Kombination aller Merkmale basierend auf Tabelle 6.3 verbessert sich die MAP von 0,0830 auf 0,1437. Dies ist zwar eine deutliche Steigerung, jedoch muss dabei beachtet werden, dass für die Filterung alle verfügbaren Bilder einer Klasse herangezogen wurden und nicht nur eine zufällig ausgewählte Obermenge der letztlich verwendeten 20 Trainingsbilder. Die Verbesserung ist jedoch ein Indiz dafür, dass möglichst saubere, d. h. Bilder, auf denen ein durchschnittliches Objekt ganzheitlich, ohne Abdeckungen und ohne extreme Lichtverhältnisse oder Schatteneffekte zu sehen ist, besser als Stichprobe für die Berechnung eines ObjectFPs geeignet sind.

7. Architektur des Pixtract-Frameworks

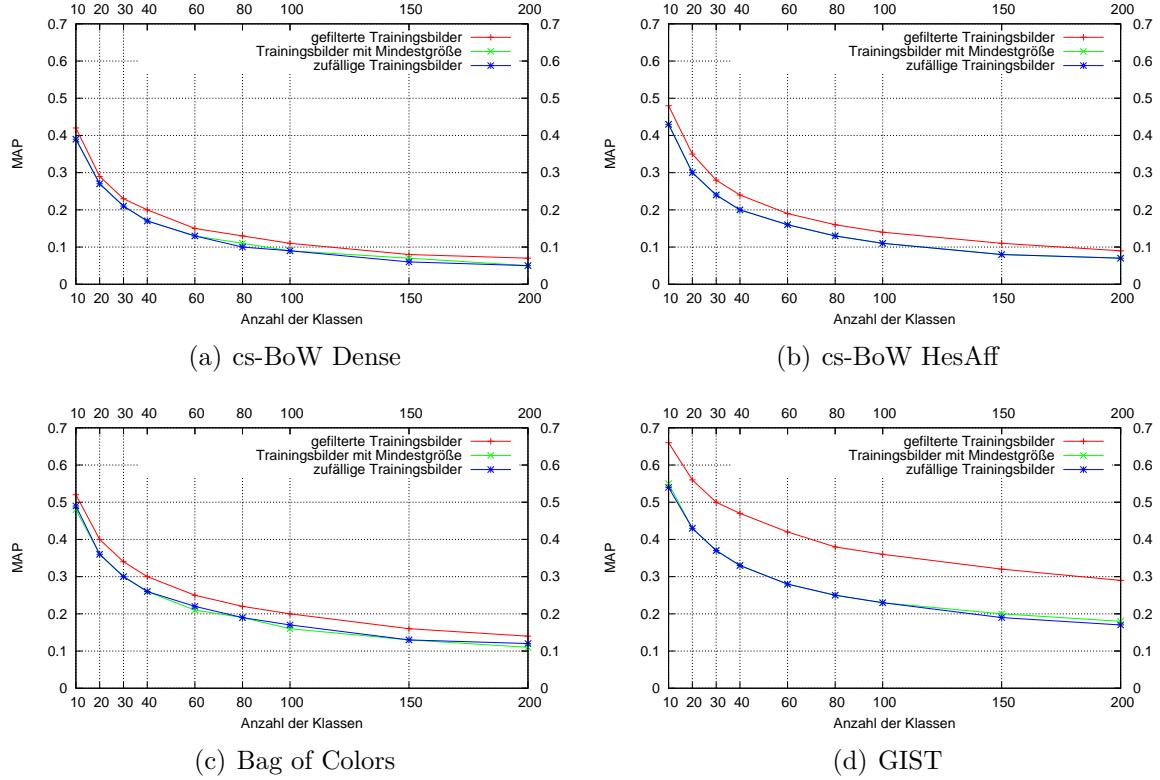


Bild 7.8: Vergleich der MAP von der Klassifikation mit ungefilterten Bildern, Bilder mit einer Mindestgröße von 128 Pixel und mit nach dem in Abschnitt 7.4.1 beschriebenen GIST-basierten Verfahren ausgewählten besten Trainingsbildern.

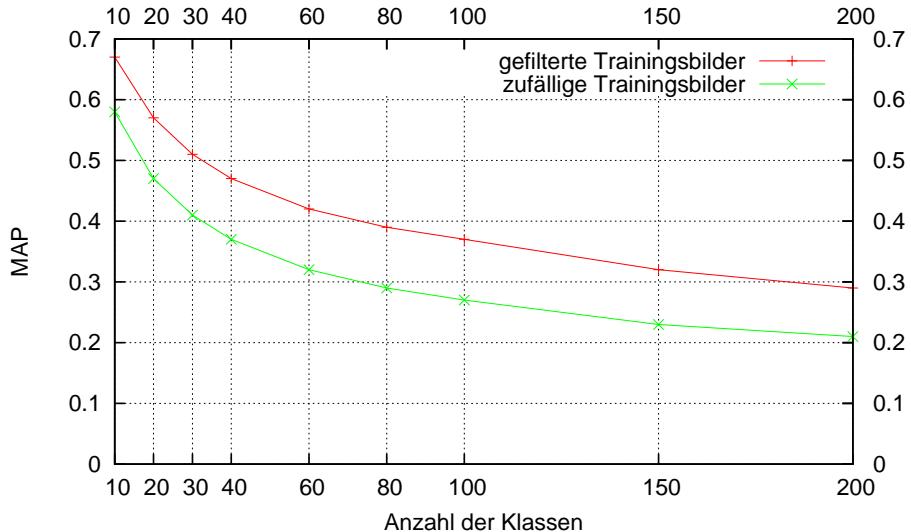


Bild 7.9: Vergleich der MAP der Kombination aller Merkmale mit und ohne Filterung der Trainingsbilder.

7.5 Zusammenfassung

In diesem Abschnitt wurde das Pixtract-Framework vorgestellt, welches das dynamische Erlernen von neuen Klassen und die Annotation von Bildern mittels der Merkmale und Verfahren aus Kapitel 5 und 6 umsetzt.

Zentraler Bestandteil von Pixtract ist das in Abschnitt 7.2.1 beschriebene konfigurierbare und durch neue Merkmalsextraktoren leicht erweiterbare Programm *FeatExt*. Der Zugriff auf Pixtract wird durch mehrere Schnittstellen ermöglicht. Die Webschnittstelle wurde in Abschnitt 7.2.2.1 vorgestellt und erlaubt die Erweiterung von Pixtract durch neue Klassen sowie die Annotation von Bildern. Mittels der in Abschnitt 7.2.2.2 beschriebenen PicID-Schnittstelle kann der Annotationsdienst von Pixtract auch in externe Bildverwaltungsprogramme oder textbasierte Suchdienste leicht integriert werden. Für die schnellere Abwicklung der Erweiterung um neue Klassen und der Annotation von Bildern wurden die Abläufe in Pixtract nach Abschnitt 7.3 weitestgehend parallelisiert. Da für die Erlernung von neuen Klassen die Stichproben von den Anwendern von Pixtract stammen, wurde in Abschnitt 7.4 zusätzlich ein einfacher Filter umgesetzt und evaluiert, welcher ungeeignete Bilder (z. B. Bilder mit Abdeckungen, Schatteneffekten) ausschließt.

Nachfolgend wird untersucht, inwieweit der in Bildern sichtbare Text für die Verbesserung der Klassifikation und Annotation verwendet werden kann.

KAPITEL 8

VERBESSERUNG DER ANNOTATION DURCH TEXT IN NATÜRLICHEN FOTOAUFNAHMEN

Die menschliche Wahrnehmung der Umgebung basiert nicht nur auf den sichtbaren Farben, Formen und Texturen, sondern wird durch zusätzliches Wissen und Erfahrungen ergänzt. Neben Objekten und Personen wird auch der im Bild enthaltene Text wahrgenommen, was zur genaueren Erkennung von Objekten beitragen kann. Diese zusätzliche Information kann sowohl bei der Klassifikation der Objekte als auch bei der nachträglichen Anreicherung der Annotation zum Einsatz kommen.

Für die Erkennung von Text können bereits etablierte OCR-Verfahren herangezogen werden. Leider sind die OCR-Verfahren zur Zeit nur auf eingescannte oder frontal abfotografierte Dokumente optimiert. In Abschnitt 8.1 werden aktuelle freie und kommerzielle OCR-Werkzeuge auf deren Eignung zur Texterkennung in natürlichen Fotoaufnahmen untersucht. Die in Pixtract eingesetzte Anreicherung der Annotation wird in Abschnitt 8.2 beschrieben. Die Erkenntnisse werden in Abschnitt 8.3 zusammengefasst.

8.1 Texterkennung in natürlichen Fotoaufnahmen

Die optische Zeichenerkennung (Optical Character Recognition (OCR)) in maschinenerstellten und handschriftlichen Dokumenten hat eine lange Vergangenheit in der Informatik. Die Entwicklungen auf diesem Gebiet waren so erfolgreich, dass nach [WLMH09] heutige OCR-Programme bei sauberen Dokumenten Zeichenerkennungsraten von mehr als 99% erreichen.

Durch die massive Verbreitung von Digitalkameras und Foto-Handys werden nach einer Schätzung von [SS11] Jahr für Jahr ca. 50 Milliarden Fotografien erstellt. Viele von diesen natürlichen Fotoaufnahmen beinhalten neben den abgebildeten Objekten und Personen auch Text. Mit der Erkennung von Text in natürlichen Fotoaufnahmen eröffnet sich ein breites Feld an möglichen Anwendungen, wie z. B.:

- Hilfe für Menschen mit Sehbehinderungen (z. B. Vorlesen von Texten, die nicht durch eine Braille-Übersetzung ergänzt wurden, wie in [LDL05] vorgeschlagen),
- mobile Anwendungen (z. B. Übersetzung von fotografiertem Text für Touristen und Ausländer, wie in [CZ09] gezeigt),
- Verbesserung der Klassifikation von Objekten in Bildern (z. B. durch multimodale Fusion von textueller und visueller Information, wie in [ZCY06] beschrieben),
- Annotation von Bildern (z. B. zur Websuche, wie in [LZ00] erläutert),
- sichtbasierte Navigation und Fahrerassistenzsysteme, wie in [WCY04] vorgeschlagen.

In den letzten Jahren ist – auch bedingt durch die Entwicklungen auf dem Gebiet der Objekterkennung – die Erkennung von Text in natürlichen Fotoaufnahmen mehr ins Rampenlicht gerückt. Natürliche Fotoaufnahmen sind im Vergleich zu eingescannten gedruckten Dokumenten weit komplexer. Nicht nur die unterschiedlichen Hintergründe, sondern auch der stark variable Text führen zu Problemen. In manchen Fällen ist es auch für Menschen schwierig die Grenze zwischen Architekturformen bzw. Schattenspielen und evtl. vorhandenem Text zu ziehen.

[LDL05] zeigt einige Unterschiede zwischen natürlichen Fotoaufnahmen und gescannten Dokumenten auf, verwendet jedoch ausschließlich Dokumente als Vorlagen und orientiert sich grundsätzlich eher an den Unterschieden zwischen Scannern und Kameras als Endgerät des Benutzers. Deswegen wurden im Rahmen der vorliegenden Arbeit die

wesentlichen Unterschiede zwischen eingescannten Dokumenten und Texten in natürlichen Fotoaufnahmen noch umfassend ermittelt und in Tabelle 8.1 dargestellt.

8.1.1 NEOCR Datensatz

Um verschiedene aktuelle OCR-Werkzeuge auf deren Eignung für die Texterkennung in natürlichen Bildern im Detail evaluieren zu können, wird ein entsprechend annotierter Datensatz benötigt. Leider sind die aktuell verfügbaren Datensätze auf diesem Gebiet nur mit Textrahmen und dem enthaltenen Text annotiert. Außerdem werden die unterschiedlichen in Tabelle 8.1 zusammengestellten Problemfelder durch die Bilder in den Datensätzen nur zum Teil abgedeckt, da z. B. vertikal angeordnete Texte komplett fehlen. Eine ausführliche Vorstellung und ein Vergleich der Datensätze ist in Abschnitt 8.1.4 zu finden.

Für eine detaillierte Auswertung bzgl. der spezifischen Problemfelder der Texterkennung in natürlichen Fotoaufnahmen sind Annotationen bzgl. der einzelnen Eigenschaften notwendig. Da keiner der vorhandenen frei zugänglichen Datensätze diese Anforderung erfüllt, wurde ein neuer Datensatz mit dem Namen Natural Environment OCR (NEOCR) erstellt [Dic11, NDMW11a, NDMW11b, NDMW12].

Im ersten Schritt wurden in Anlehnung an die Eigenschaften von natürlichen Fotoaufnahmen aus Tabelle 8.1 die zu annotierenden Metadaten definiert. Diese Metadaten werden in Abschnitt 8.1.2 und Abschnitt 8.1.3 genauer vorgestellt. Die Grundmenge an Bildern bildeten Aufnahmen von Mitarbeitern des Lehrstuhls. Die Aufnahmen wurden mit unterschiedlichen Digitalkameras und Einstellungen erstellt, wodurch eine natürliche Diversität der Bilder bzgl. der Bildeigenschaften erreicht wurde. Anschließend wurden insgesamt 659 Bilder mit Textinhalt manuell selektiert, wobei die einzelnen Dimensionen durch mindestens 100 Ausschnitte abgedeckt wurden.

Für die Annotation der Bilder wurde das webbasierte Werkzeug LabelMe¹ von [RTMF08] verwendet, welches wegen der speziellen Angaben für den NEOCR Datensatz (z. B. Verzerrungsbox) zusätzlich angepasst wurde. Die Annotationen wurden über eine Webschnittstelle eingegeben bzw. soweit möglich durch ImageMagick² automatisch berechnet und anschließend als XML-Datei abgespeichert. Die 659 Bilder wurden mit

¹ <http://labelme.csail.mit.edu>

² <http://www.imagemagick.org>

EIGENSCHAFT	GESCANNTES DOKUMENT	NÄTÜRLICHE FOTOAUFNAHME
Abdeckung	keine	horizontal, vertikal oder willkürlich kombiniert
Anordnung der Zeichen	gerade, horizontale Zeilen	horizontale und vertikale Linien, gekrümt, wellenförmig
Farben	meistens schwarzer Text auf weißem Hintergrund	hohe Farbvarianz, auch heller Text auf dunklem Hintergrund oder nur leichte Farbunterschiede (z. B. bei Gravuren)
Hintergrund	gleichmäßig, überwiegend weißes bzw. helles Papier	verschiedenfarbig, auch dunkel oder texturiert
Kameraposition	fest, Dokument auf Glasplatte	extrem variabel, Verzerrungen sind allgegenwärtig
Kontrast	sehr hoch (schwarzer/dunkler Text auf weißem/hellem Untergrund)	abhängig von Farbe, Schattengebung, Beleuchtung, Textur
Oberfläche	Text auf glattem Papier	Text auf Objekten mit nicht ebener Oberfläche, diverse Verzerrungen
Rauschen	beschränkt/vernachlässigbar	Beleuchtung, Reflexionen, Sensorrauschen
Rotation	horizontal ausgerichteter Text, bzw. rotiert $\pm 90^\circ$	beliebige Rotationswinkel
Schriftgröße	beschränkte Auswahl	hohe Vielfalt
Schrifttyp (Dokument)	üblicherweise ein bis zwei Schrifttypen	viele Schrifttypen
Schrifttyp (generell)	Maschinen-/Handschrift	Maschinen-/Handschrift, aber auch spezielle Schriften (z. B. zusammengesetzt aus Glühlampen)
Schärfe	scharf	Bewegungsunschärfe, Unschärfe durch Schärfentiefe
Zeilenanzahl	mehrere Textzeilen	oft nur einzelne Wörter bzw. einzelne Zeilen

Tabelle 8.1: Unterschiede zwischen Text in eingescannten Dokumenten und Text in natürlichen Fotoaufnahmen.

insgesamt 5 238 Textfeldern annotiert. Dabei wurde akribisch darauf geachtet, dass jeder im Bild erkennbare Text annotiert wurde.

In den folgenden Abschnitten werden die annotierten globalen und lokalen Metadaten vorgestellt. Bei der Annotation wurde darauf geachtet so viel Information wie nur möglich automatisch zu ermitteln. Abbildung 8.1 zeigt Beispielbilder aus dem NEOCR Datensatz, welche typische Probleme bei der Texterkennung in natürlichen Fotoaufnahmen veranschaulichen.

8.1.2 Globale Metadaten

Für jedes Bild werden global der Dateiname, Ordner, Quelle und Bildeigenschaften angegeben. Letzteres umfasst die Höhe und Breite des Bildes, Farbtiefe, Helligkeit und Kontrast. Die Helligkeit entspricht dem Mittelwert, der Kontrast der Standardabweichung des Helligkeitskanals (Y-Kanal) des YUV-Farbmodells. Beide Werte werden automatisch durch ImageMagick ermittelt.

8.1.3 Lokale Metadaten

Alle in den Bildern sichtbaren Wörter und zusammenhängenden Texte wurden durch seitenparallele Rechtecke markiert und der enthaltene Text annotiert. Zusammenhängende Texte entsprechen dabei mehreren Wörtern und ggf. mehreren Zeilen Text in gleicher Schriftart, Schriftgröße und Hintergrund. Um Textbereiche präziser umranden zu können und Fehlern aus überlappenden Rechtecken entgegen zu wirken, wurden zusätzlich zu den seitenparallelen Rechtecken Verzerrungsvierecke annotiert. Die annotierten Metadaten für die einzelnen Textbereiche wurden in optische, geometrische und typografische Metadaten eingeordnet.

8.1.3.1 Optische Eigenschaften

Zu optischen Eigenschaften gehören Informationen über die Textur, Helligkeit, Kontrast, Inversion, Auflösung, Rauschen und Unschärfe des Bildausschnittes.



Bild 8.1: Beispiele aus dem NEOCR Datensatz, welche typische Problemfelder der Texterkennung in natürlichen Fotoaufnahmen aufzeigen.

Textur

Die Textur eines Bildausschnittes ist automatisch schwierig zu ermitteln, da Unterschiede in der Textur den Text bilden können und auch der Text selber die Textur sein kann. Aus diesem Grund wurden drei Stufen für die Angabe der Textur definiert und die Bilder manuell den Kategorien zugeordnet:

- niedrig (low): einfarbiger Text auf einfarbigem Hintergrund,
- mittel (mid): mehrfarbiger Text oder mehrfarbiger Hintergrund,
- hoch (high): mehrfarbiger Text und mehrfarbiger Hintergrund, oder Text mit einer nicht durchgängigen Fläche (z. B. aus einzelnen Glühbirnen bestehende Leuchtreklamen, wie in Abbildung 8.1(f) dargestellt).

Helligkeit und Kontrast

Helligkeit und Kontrast werden ähnlich zu Abschnitt 8.1.2 aus dem Mittelwert bzw. der Standardabweichung des Y-Kanals des YUV-Farbmodells automatisch durch ImageMagick für den Bildausschnitt ermittelt. Zusätzlich wird beim Kontrast auch noch die

Inversion annotiert. Durch die Inversion wird angegeben ob es sich um dunklen Text auf hellem Hintergrund oder hellem Text auf dunklem Hintergrund handelt.

Auflösung

Im Unterschied zu hochauflösenden Scannern mit bis zu 1000dpi erreichen mit Digitalkameras erstellte Bilder höchstens 300dpi. Zusätzlich wirkt sich auch die Brennweite auf die resultierende Aufnahme aus. Je kleiner die Brennweite, desto größer ist der Bereich, der von der Linse auf den gleich großen Sensor projiziert wird. Abhängig von der Pixeldichte und der Größe des Kamerabildsensors kann ein kleiner Text unkenntlich werden. Aus diesem Grund wurde die Auflösung in den Metadaten als die Anzahl der Pixel im Rechteck geteilt durch die Anzahl der annotierten Zeichen, also als Anzahl der Pixel je Zeichen erfasst.

Rauschen

Bildrauschen entsteht einerseits durch die Rauschempfindlichkeit des Kamerabildsensors und andererseits durch Artefakte des verwendeten Bildkompressionsverfahrens (z. B. bei JPEG-Bildern). Rauschen verstärkt sich bei höherer ISO-Empfindlichkeit und höheren Kompressionsraten. Da Rauschen und Textur nur sehr schwierig auseinander zu halten sind, wurde das Bildrauschen nach Augenmaß in die Kategorien niedrig (low), mittel (mid) und hoch (high) eingestuft.

Unschärfe

Bei der Unschärfe von Bildern kann generell zwischen Tiefunschärfe und Bewegungsunschärfe unterschieden werden.

Tiefunschärfe verstärkt sich durch die Verwendung von kleineren Blenden, d. h. der Verwendung einer größeren Linsenöffnung. Die Auswirkung der Tiefunschärfe auf das abgebildete Objekt ist dabei abhängig von der Brennweite und dem Bildfokus. Ähnliche Unschärfeeffekte können auch bei der Bildkompression auftreten. Bewegungsunschärfe entsteht durch Verwacklungen durch den Fotografen oder durch Objekte im Bild, die sich in Bewegung befinden.

Eine Übersicht für die Messung von Bildunschärfe ohne Referenzbild ist in [FK09] zu finden. Da die beschriebenen besten Verfahren leider mit hohem Berechnungsaufwand verbunden sind, wird beim NEOCR-Datensatz für die Unschärfe eines Bildausschnittes

eine Kombination verschiedener weniger rechenintensiver Verfahren verwendet. Zuerst werden mit einem Laplacian of Gaussian (LoG) Filter die Kanten detektiert. Für das ermittelte Kantenbild wird eine Fourier-Transformation durchgeführt und anschließend die Steilheit (kurtosis) der berechneten Spektralanalyse ermittelt. Ein hoher Wert steht dabei für ein großes Maß an Unschärfe.

8.1.3.2 Geometrische Eigenschaften

Unter geometrischen Eigenschaften werden die Verzerrung, die Rotation und die Abdeckung des Textes sowie die Anordnung der Zeichen erfasst.

Verzerrung

Bei den meisten natürlichen Fotoaufnahmen ist die Ebene des Bildsensors nicht parallel zu der fotografierten Textebene. Das führt dazu, dass der fotografierte Text auf dem Foto perspektivisch verzerrt erscheint. Zur Annotation der Verzerrung sind mehrere Möglichkeiten denkbar. Im NEOCR Datensatz wurden 8 Fließkommawerte in Anlehnung an [Wol94] annotiert. Die Werte können als folgende Matrix repräsentiert werden:

$$\begin{pmatrix} s_x & r_x & p_x \\ r_y & s_y & p_y \\ t_x & t_y & 1 \end{pmatrix},$$

wobei s_x und s_y für die Skalierung, r_x und r_y für die Rotation, t_x und t_y für die Translation und p_x und p_y für die perspektivische Verzerrung stehen.

In [Wol94] wurden die Gleichungen zur Berechnung des Verzerrungsvierecks für Begrenzungsrahmen mit Einheitslänge definiert. Die Herleitung der angepassten Formel für beliebige Längen inklusive einer genaueren Interpretation der Werte ist in Anhang D.1 zu finden. Die Punkte des verzerrten Vierecks lassen sich anhand der Matrix und der Koordinaten des Ursprungsrechtecks folgendermaßen berechnen:

$$x' = \frac{s_x x + r_y y + t_x}{p_x x + p_y y + 1} \quad y' = \frac{r_x x + s_y y + t_y}{p_x x + p_y y + 1} \quad (8.1)$$

Rotation

Im Gegensatz zu eingescannten Dokumenten können Texte in natürlichen Fotoaufnahmen wegen der freien Positionierung und Drehung der Kamera beliebig rotiert auftreten. Die Rotation wird automatisch aus dem Verzerrungspolygon berechnet. Die genaue Berechnung ist in Anhang D.2 erläutert. Im Datensatz ist die Abweichung von der horizontalen Achse des gesamten Bildes als Gradmaß angegeben.

Anordnung

In natürlichen Fotoaufnahmen können Zeichen auch vertikal angeordnet sein, wie es z. B. häufig bei Hotelschildern zu sehen ist. Einige Texte der realen Welt folgen auch gebogenen Linien. Bei den Textfeldern werden horizontal, vertikal und zirkular angeordnete Texte unterschieden. Einzeln auftretende Zeichen wurden als horizontal angeordnet annotiert.

Abdeckung

Abhängig von der Wahl des Ausschnitts und den Objekten im Bild kann ein Text zum Teil abgeschnitten oder überdeckt sein. In den Annotationen werden die horizontale und vertikale Abdeckung unterschieden. Zusätzlich wurde auch der prozentuale Anteil der Abdeckung annotiert.

8.1.3.3 Typografische Eigenschaften

Unter typografischen Eigenschaften werden Metadaten zur Schriftart und Sprache subsummiert.

Schriftart

Die Schriftart des umrahmten Texts wurde manuell in die Kategorien Druck (print), Handschrift (handwriting) und Spezial (special) eingestuft. Beim annotierten Text wird Groß- und Kleinschreibung unterschieden. Die Schriftgröße kann aus der Umrahmung und der Auflösung errechnet werden. Dicke und Kursivität der Schrift wurden nicht annotiert.

Sprache

Die Angabe der Sprache kann ein wesentliches Indiz sein, sofern zur Verbesserung der Texterkennung oder zur Korrektur des erkannten Texts Vokabulare eingesetzt werden. Da die Bilder des Datensatzes in unterschiedlichen Ländern erstellt worden sind, beinhaltet der NEOCR-Datensatz Texte in insgesamt 15 verschiedenen Sprachen. Der Zeichenvorrat beschränkt sich dabei jedoch auf lateinische Zeichen. Bei einigen Texten kann die Sprache nicht eindeutig ermittelt werden. Für diese Spezialfälle wurden die zusätzlichen Sprachkategorien für Zahlen, Abkürzungen und Firmennamen eingeführt.

8.1.3.4 Zusammenfassung

Das bereits vorhandene XML-Schema von LabelMe wurde durch die oben eingeführten Metadaten angepasst und erweitert. Das angepasste XML-Schema für die Annotation von Texten in natürlichen Fotoaufnahmen mittels LabelMe ist in Anhang E.3 zu finden. Das Attribut *difficult* wurde aus dem ursprünglichen XML-Schema der Annotationssoftware übernommen. Als *difficult* wurden diejenigen Texte eingestuft, welche für Menschen besonders schwer lesbar und ohne Kenntnisse über den Kontext nicht erkennbar sind, wie z. B. Graffitis, besonders stark verrauschte Umgebungen oder großflächig abgedeckter Text.

Abbildung 8.2 zeigt ein Beispielbild mit der angepassten LabelMe-Webschnittstelle. Die zugehörige Annotation basierend auf den oben definierten Metadaten ist in Tabelle 8.2 (bzw. als XML in Anhang E.4) dargestellt.

In Abbildung 8.3 sind Statistiken für ausgewählte Dimensionen des NEOCR-Datensatzes dargestellt. Die Grafiken verdeutlichen die hohe Diversität der Bilder im Datensatz. Die in NEOCR zusätzlich annotierten Metadaten ermöglichen einen genaueren und detaillierteren Vergleich zwischen unterschiedlichen Ansätzen für OCR in natürlichen Fotoaufnahmen und die präzisere Identifikation von Lücken und Problembereichen bei der Texterkennung.

8.1.4 Vergleich mit anderen Datensätzen

Öffentlich verfügbare Datensätze für die Texterkennung in natürlichen Fotoaufnahmen sind rar. Nachfolgend werden speziell für diesen Zweck entstandene Datensätze kurz vorgestellt und analysiert.

EIGENSCHAFT	DATENTYP	WERTEBEREICH	WERT
Textur	string	low, mid, high	mid
Helligkeit	float	0 - 255	164,493
Kontrast	float	0 - 127	36,6992
Invertierung	boolean	true, false	false
Auflösung	float	1 - 2 000 000	49 810
Rauschen	string	low, mid, high	low
Unschärfe	float	0 - 100 000	231,787
Verzerrung	8 floats	sx: [-1;5], sy: [-1;1,5], rx: [-15;22], ry: [-23;4], tx: [0;1505], ty: [0;1419], px: [-0,03;0,07], py: [-0,02;0,02]	sx: 0,92, sy: 0,67, rx: -0,04, ry: 0, tx: 0, ty: 92, px: -3,28-05, py: 0 py: [-0,02;0,02]
Rotation	float	0 - 360	2,00934289847729
Anordnung	string	horizontal, vertical, circular	horizontal
Abdeckung	integer	0 - 100	5
Abdeckungsart	string	horizontal, vertical	vertical
Schriftart	string	standard, handwriting	special, standard
Sprache	string	german, english, spanish, german hungarian, italian, latin, french, belgian, russian, tur- kish, greek, swedish, czech, portoguese, numbers, roman date, abbreviation, company, person, unknown	german
Schwierigkeit	boolean	true, false	false

Tabelle 8.2: Datentypen und Wertebereiche der Metadaten sowie Beispiel-Annotationswerte für Abbildung 8.2.



Bild 8.2: Beispielbild „Finstere Gasse“ in der angepassten Annotationssoftware LabelMe. Das blaue Viereck kennzeichnet den seitenparallelen Textrahmen, während mit türkiser Farbe das Verzerrungsviereck dargestellt ist.

Am weitesten in der Forschung verbreitet ist der ICDAR-2003-Robust-Reading¹-Datensatz [LPS⁺03, LPS⁺05]. Er beinhaltet 258 Trainings- und 251 Testbilder, alle annotiert mit insgesamt 2 263 Textrahmen. Die Textrahmen sind alle parallel zu den Seiten des Bildes, was bei natürlichen Fotoaufnahmen mit häufig auftretendem verzerrten und rotierten Text unvorteilhaft ist und wodurch die Bilder gestellt und künstlich wirken. Die Schriftarten weisen zwar eine relativ hohe Diversität auf, jedoch liegt der Fokus beim überwiegenden Teil der Fotos auf den Texten. Der Datensatz beinhaltet vorwiegend Innenaufnahmen, darunter auch meistens Bücher oder Nahaufnahmen von Markennamen auf Geräten. Im Datensatz ist kein einziges Bild vorhanden, in dem die Zeichen des fotografierten Textes vertikal oder zirkular angeordnet wären. Die hohe Variabilität bei natürlichen Fotoaufnahmen, wie z. B. Schatten, Beleuchtung oder Zeichenanordnung, wird vom Datensatz nicht abgedeckt.

¹ <http://algoval.essex.ac.uk/icdar/Datasets.html>

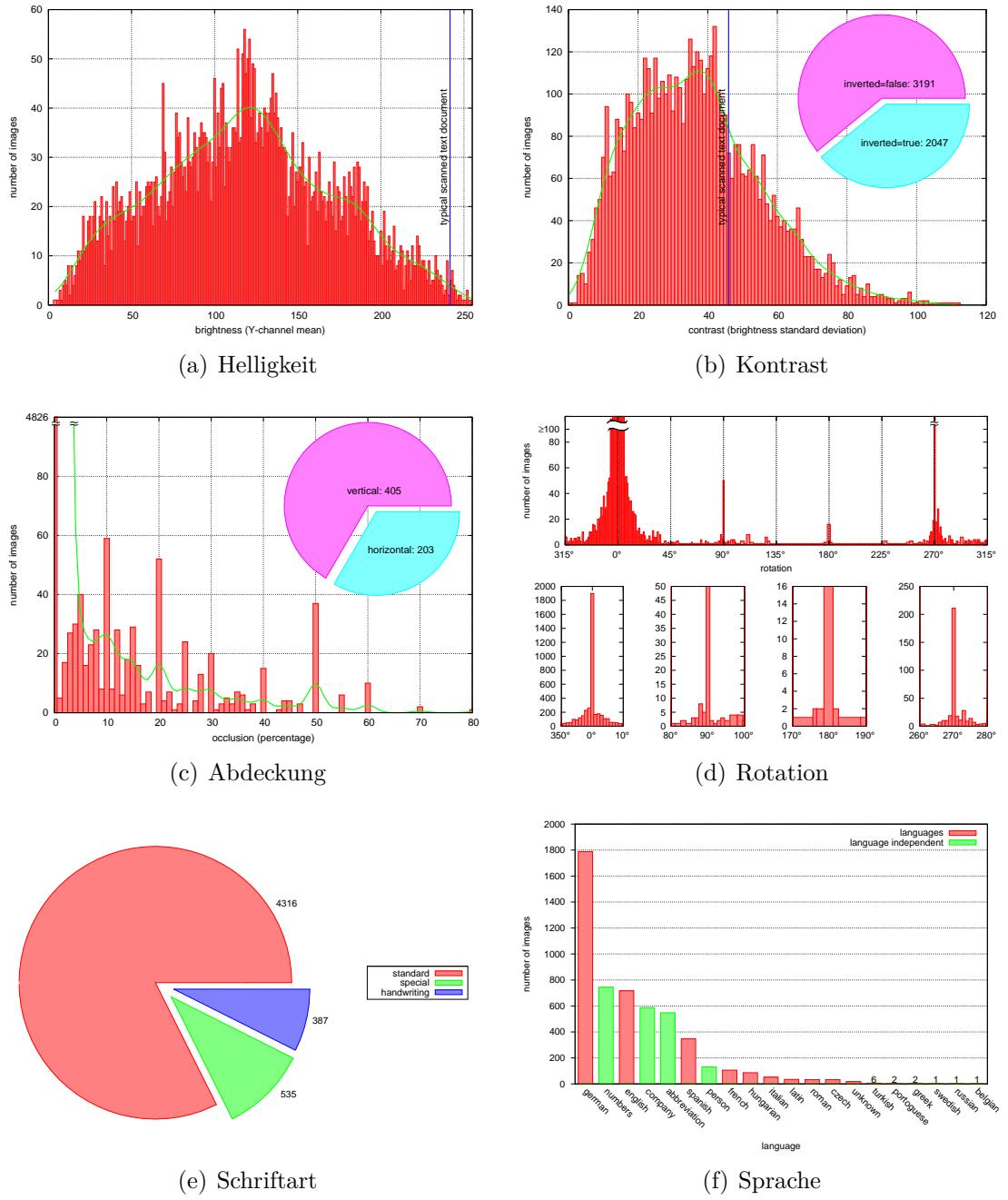


Bild 8.3: Statistiken zu Helligkeit, Kontrast, Rotation, Abdeckung, Schriftart und Sprache, welche die hohe Diversität des NEOCR-Datensatzes bzgl. der Eigenschaften von Texten in natürlichen Fotoaufnahmen bestätigen (siehe Tabelle 8.1). Die Bilder 8.3(a) und 8.3(b) zeigen zusätzlich den Wert für eine gescannte Seite aus einem Textbuch. Die Anzahl der Bilder bezieht sich bei allen Statistiken auf die Anzahl der Textrahmen.

Der Chars74K¹-Datensatz wurde in [CBV09] vorgestellt und beinhaltet natürliche Fotoaufnahmen mit lateinischen und indischen Zeichen. Die meisten der insgesamt 1922 Fotos zeigen Schilder, Plakate und Werbung aus einem frontalen Blickwinkel. 900 Bilder davon sind mit Textrahmen annotiert, wovon lediglich 312 Bilder lateinische Zeichen enthalten. Leider wurden Fotos mit Abdeckung, niedriger Auslösung oder Rauschen explizit aus dem Datensatz entfernt. Außerdem sind auch nicht alle sichtbaren Texte in einem Bild annotiert, was die Auswertungen auf ganzen Bildern unmöglich macht.

In [WB10] wurde der Street-View-Text-Datensatz² vorgestellt, welcher auf Bildern von Google Street View³ aufbaut. Die 350 Bilder des Datensatzes zeigen größtenteils Außenaufnahmen von Firmenschildern. Insgesamt wurden 904 Textrahmen annotiert. Leider sind die Rahmen ähnlich zum ICDAR-2003-Datensatz seitenparallel, was für die genaue Markierung von Texten in natürlichen Bildern aufgrund ihrer hohen Variabilität bzgl. Verzerrung und Rotation nicht ausreichend ist. Ein weiterer Nachteil beim Datensatz ist, dass nicht alle sichtbaren Texte in den Bildern annotiert wurden, was Evaluationen auf ganzen Bildern erheblich erschwert.

In [EOW10] wurde ein neues Verfahren zur Texterkennung in natürlichen Szenen basierend auf Strichdicken vorgestellt. Der Algorithmus wurde auf dem ICDAR-2003- und zusätzlich auf einem neu erstellten eigenen Datensatz evaluiert. Die insgesamt 307 annotierten Bilder des Microsoft-Text-Detection-Datensatzes⁴ decken die Eigenschaften von natürlichen Fotoaufnahmen besser ab als der ICDAR-Datensatz. Leider wurden jedoch nicht alle im Bild sichtbaren Texte annotiert und die Textrahmen sind alle seitenparallel.

Weiterhin existieren noch spezielle Datensätze für die Erkennung von Nummernschildern, Büchern oder Zahlen. Keiner der in diesem Abschnitt vorgestellten Datensätze deckt das volle Spektrum der Diversität von natürlichen Fotoaufnahmen ab. Auch die Annotationen beschränken sich leider nur auf den beinhalteten Text bzw. den Rahmen, in dem sich der Text befindet. Zusätzliche Metadaten wären für den Vergleich von verschiedenen OCR-Ansätzen nützlich und würden auch bei der Identifikation von Problemen bei einzelnen Verfahren sehr hilfreich sein.

1 <http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/>

2 <http://vision.ucsd.edu/~kai/svt/>

3 <http://maps.google.com>

4 http://research.microsoft.com/en-us/um/people/eyalofek/text_detection_database.zip

Da die Datensätze in diesem Abschnitt lediglich die Koordinaten der Rahmen und den beinhalteten Text für die Bilder annotieren, beschränkt sich der Vergleich mit dem NEOCR-Datensatz auf die Anzahl der annotierten Bilder, die Anzahl der Textrahmen und die durchschnittliche Anzahl an Zeichen je Textrahmen. Beim Chars74K Datensatz wurden nur Annotationen mit lateinischen Zeichen oder Zahlen berücksichtigt. Da bei diesem Datensatz neben ganzen Wörtern auch zusätzlich alle einzelnen Zeichen annotiert wurden, werden beim Vergleich nur Zeichenketten mit einer Länge größer 1 berücksichtigt. Der Vergleich zwischen den einzelnen Datensätzen ist in Tabelle 8.3 dargestellt.

DATENSATZ	ANZAHL BILDER	ANZAHL TEXTRAHMEN	DURCHSCHN. ANZAHL ZEICHEN JE RAHMEN
ICDAR 2003	509	2 263	6,15
Chars74K	312	2 112	6,47
MS Text Detection	307	1 729	10,76
Street View Text	350	904	6,83
NEOCR	659	5 238	17,62

Tabelle 8.3: Vergleich verschiedener Datensätze für die Texterkennung in natürlichen Fotoaufnahmen.

Verglichen mit anderen Datensätzen für die Texterkennung in natürlichen Fotoaufnahmen bietet der NEOCR Datensatz weit mehr annotierte Textrahmen. Da bei NEOCR nicht nur Wörter, sondern auch ganze zusammenhängende Texte annotiert wurden, ist die durchschnittliche Anzahl der Zeichen je Textrahmen deutlich höher. Keiner der verwandten Datensätze ist mit zusätzlichen Metadaten angereichert, was detailliertere Auswertungen ermöglichen und Schwächen von einzelnen OCR-Ansätzen besser aufdecken könnte.

8.1.5 Distanzmaße für den Vergleich von Zeichensequenzen

Für die Bewertung der Texterkennung von aktueller OCR-Software sind Distanzmaße für den Vergleich zwischen annotiertem und erkanntem Text nötig. Von den OCR-Anwendungen werden zum Teil Zeichen verwechselt oder zusätzliche Zeichen erkannt, welche entsprechend von den Distanzmaßen berücksichtigt werden müssen.

Einen guten Überblick über verschiedene Distanzmaße für den Vergleich von Zeichenketten bieten [Kru83] und [Nav01]. Nachfolgend werden die bekanntesten und am häufigsten eingesetzten Distanzmaße kurz vorgestellt.

8.1.5.1 Allgemeine Definitionen

Für die einheitliche Beschreibung und den einfacheren Vergleich der verschiedenen Distanzmaße werden zuerst allgemeine Definitionen festgelegt. Die Distanz $d(x_n, y_m)$ zwischen zwei beliebigen Zeichenketten oder Wörtern x_n und y_m ist definiert als der minimale Kostenaufwand, welcher für die Umwandlung einer Zeichenkette x_n in y_m nötig ist. \emptyset bezeichnet hierbei das leere Wort und n entspricht der Länge der Zeichenkette x_n . Aus mathematischer Sicht müssen die Distanzfunktionen folgende vier Axiome erfüllen:

- Nichtnegativität: $d(x_n, y_m) \geq 0$,
- Nullwert: $d(x_n, y_m) = 0$ falls $x_n = y_m$,
- Dreiecksungleichheit: $d(x_n, y_m) + d(y_m, z_o) \geq d(x_n, z_o)$,
- Symmetrie: $d(x_n, y_m) = d(y_m, x_n)$.

Alle Zeichenketten x_n und y_m erfüllen die ersten drei Eigenschaften, wobei durch zusätzliche Einhaltung der Symmetrieeigenschaft für Operationen von den vorliegenden Zeichenfolgen ein metrischer Raum aufgespannt wird.

Operationen zur Umwandlung werden als $\delta(a, b) = t$ definiert, wobei a und b unterschiedliche Zeichen sind und t die Kosten für die Umwandlung angibt. [Kru83] und [Nav01] beschränken die Methoden zur Umwandlung von Zeichenketten auf folgende Operationen:

- Einfügung: $\delta(\emptyset, a)$, Zeichen a einfügen,
- Löschung: $\delta(a, \emptyset)$, Zeichen a löschen,
- Ersetzung: $\delta(a, b)$ für $a \neq b$, Zeichen a durch b ersetzen,
- Vertauschung: $\delta(ab, ba)$ für $a \neq b$, benachbarte Zeichen a und b austauschen.

Zusätzlich wird in [Kru83] noch die Verschmelzung mehrerer Buchstaben zu einem einzelnen Zeichen sowie die Erweiterung als inverse Operation beschrieben. Diese Operationen spielen jedoch beim Vergleich zwischen Zeichenketten in der vorliegenden Auswertung eine untergeordnete Rolle und werden deshalb nicht weiter erläutert.

8.1.5.2 Hamming-Abstand

Als einfachstes Distanzmaß können die Zeichen einer Zeichenkette exakt verglichen werden. Dieses Distanzmaß ist unter dem Namen Hamming-Distanz [Ham50] bekannt.

Beim Vergleich von zwei Zeichenketten wird die Anzahl k der unterschiedlichen Zeichen zusammengerechnet, wobei jedes anweichende Zeichen mit Kosten 1 verbunden ist. Im Sinne der allgemeinen Definition aus Abschnitt 8.1.5.1 kann der Hamming-Abstand als Umwandlung einer Zeichenkette x_n in y_m mit k Ersetzungen aufgefasst werden. Zur Berechnung der Hamming-Distanz wird die Anzahl bestimmt, in wie vielen Positionen sich zwei Zeichenketten unterscheiden:

$$d(x_n, y_m) = \sum_{x_n[i] \neq y_m[i]} 1, \quad (8.2)$$

wobei $x_n[i]$ das Zeichen an Position i in der Zeichenkette x_n ist. Sofern $n \neq m$, werden die übrigen Zeichen $|n - m|$ als unterschiedliche Zeichen aufgefasst und zum Abstand addiert.

8.1.5.3 Levenshtein-Distanz

Die Levenshtein- [Lev66] bzw. Edit-Distanz genannte Funktion ist das am häufigsten eingesetzte Distanzmaß für den Vergleich von Zeichenketten. Im Wesentlichen kann die Levenshtein-Distanz als ein Vergleich mit k Unterschieden zwischen zwei Zeichenketten aufgefasst werden, wobei Einfügungen sowie das Löschen und Ersetzen einzelner Zeichen als Operationen erlaubt sind. Sofern die Kosten für diese Transformationen auf 1 gesetzt werden, kann die Levenshtein-Distanz auch als die minimale Anzahl an nötigen Transformationen betrachtet werden, um die Zeichenkette x_n in y_m zu überführen. Formal ergibt sich folgende Definition:

$$d(\emptyset, \emptyset) = 0 \quad (8.3)$$

$$d(x_n, \emptyset) = d(\emptyset, x_n) = n \quad (8.4)$$

$$d(x_n, y_m) = \min \begin{cases} d(x_{n-1}, y_{m-1}) + 0 & \text{falls } x_n[i] = y_m[j], \\ d(x_{n-1}, y_{m-1}) + 1 & (\text{Ersetzung}), \\ d(x_n, y_{m-1}) + 1 & (\text{Einfügung}), \\ d(x_{n-1}, y_m) + 1 & (\text{Löschen}), \end{cases} \quad (8.5)$$

wobei x_{n-1} der um ein Zeichen gekürzten Zeichenkette x_n entspricht.

Die Damerau-Levenshtein-Distanz von [Dam64] erweitert die Levenshtein-Distanz um eine Operation zur Vertauschung einzelner Zeichen. Für die Vertauschungsoperation werden Kosten c berechnet. Die Definition der Levenshtein-Distanz aus Gleichung 8.5 bedarf hierzu der folgenden Anpassung:

$$d(x_n, y_m) = \min \begin{cases} d(x_{n-1}, y_{m-1}) + 0 & \text{falls } x_n[i] = y_m[j], \\ d(x_{n-1}, y_{m-1}) + 1 & (\text{Ersetzung}), \\ d(x_n, y_{m-1}) + 1 & (\text{Einfügung}), \\ d(x_{n-1}, y_m) + 1 & (\text{Löschen}), \\ d(x_{n-2}, y_{m-2}) + c & (\text{Vertauschung}) \\ & \text{falls } x_n[i] = y_m[i-1] \text{ oder } x_n[i-1] = y_m[i]. \end{cases} \quad (8.6)$$

8.1.5.4 Longest Common Substring

[NW70] und [RLG⁺05] definieren den Longest Common Substring (LCS) Abstand als die längste gemeinsame Folge von Zeichen in den Zeichenketten x_n und y_m , unter der Bedingung, dass die Zeichen in der selben Reihenfolge in beiden Zeichenketten erscheinen. Mittels folgender rekursiver Definition kann die Länge dieser gemeinsamen Zeichenkette ermittelt werden:

$$lcs(\emptyset, \emptyset) = lcs(x_n, \emptyset) = lcs(\emptyset, x_n) = 0 \quad (8.7)$$

$$lcs(x_n, y_m) = \begin{cases} lcs(x_{n-1}, y_{m-1}) + 1 & \text{falls } x_n[i] = y_m[j], \\ \max(lcs(x_{n-1}, y_m), lcs(x_n, y_{m-1})) & \text{falls } x_n[i] \neq y_m[j], \end{cases} \quad (8.8)$$

wobei das Löschen nur am Ende der Zeichenketten erlaubt ist. Ausgehend von der Länge des LCS kann die Distanz zwischen Zeichenketten x_n und y_m mit Längen n bzw. m nach [BHR00] folgendermaßen berechnet werden:

$$d(x_n, y_m) = n + m - 2lcs(x_n, y_m). \quad (8.9)$$

Effiziente Implementierungen des LCS-Algorithmus werden in [AG87, Hir75] und [BHR00] vorgestellt.

8.1.5.5 Jaro-Distanz

Die Jaro-Distanz wird nach [Jar89] und [Win90] als die gewichtete Summe der übereinstimmenden Zeichen und der Anzahl an Vertauschungen berechnet:

$$d(x_n, y_m) = \left(\frac{w_x c}{n} + \frac{w_y c}{m} + \frac{w_\tau(c - \tau)}{c} \right), \quad (8.10)$$

wobei w_x und w_y , die mit x_n und y_m verbundenen Gewichte und w_τ das Gewicht der Vertauschungen bezeichnet. Zwei Zeichen werden als übereinstimmend interpretiert, wenn sie nicht weiter als $c = \frac{\max(n, m)}{2} - 1$ auseinander liegen. Die Anzahl an nötigen Vertauschungen τ wird durch positionsweises Vergleichen der gemeinsamen Zeichen berechnet. Dabei entspricht die Hälfte der Anzahl an nicht übereinstimmenden Zeichen der Anzahl an Vertauschungen.

Die Jaro-Winkler-Distanz erweitert die Jaro-Distanz durch die zusätzliche Überprüfung der Übereinstimmung zwischen den ersten Buchstaben in den Zeichenketten x_n und y_m . Nach [Win90] wird die Jaro-Winkler-Distanz folgendermaßen definiert:

$$d(x_n, y_m) = d_J(x_n, y_m) + \gamma \cdot 0,1(1 - d_J(x_n, y_m)), \quad (8.11)$$

wobei $d_J(x_n, y_m)$ der Jaro-Distanz zwischen x_n und y_m aus Gleichung 8.10 entspricht und γ die Länge des gemeinsamen Präfix bis zu einem Maximalwert von 4 bezeichnet. Als Gewicht für w_x , w_y und w_τ wurde in [Win90] 1/3 angegeben, was auch für die Evaluation in Abschnitt 8.1.6 verwendet wurde.

8.1.5.6 Normalisierte Ähnlichkeit zweier Zeichenketten

Die Hamming-, Levenshtein-, Damerau-Levenshtein- und LCS-Distanzen berechnen jeweils die Zahl der Operationen, die zur Überführung der Zeichenkette x_n in y_m nötig ist. Da die Zahl der Transformationen somit direkt von der Länge der Zeichenketten abhängig ist, müssen für die Auswertungen in Abschnitt 8.1.6 die Distanzmaße normalisiert werden. Hierfür werden die Distanzen durch die Anzahl der Zeichen der längeren Zeichenkette dividiert. Dadurch ergeben sich Distanzen im Wertebereich zwischen 0 und 1.

Für die Ermittlung der normalisierten Ähnlichkeit von zwei Zeichenketten wird die normalisierte Distanz von 1 subtrahiert. Im Prinzip entspricht dieses Ähnlichkeitsmaß der prozentualen Übereinstimmung zwischen Zeichenketten. Die normalisierte Ähnlichkeit zweier Zeichenfolgen x_n und y_m mit der Länge n bzw. m wird somit folgendermaßen definiert:

$$s(x_n, y_m) = 1 - \frac{d(x_n, y_m)}{\max(n, m)}. \quad (8.12)$$

8.1.5.7 Zusammenfassung

In diesem Abschnitt wurden die am häufigsten verwendeten Distanzfunktionen zum Vergleich von Zeichenketten kurz vorgestellt. Zusätzlich wurde ein normalisiertes Ähnlichkeitsmaß für Zeichenketten definiert.

OCR-Anwendungen werden meistens zur Erkennung von Texten in eingescannten Dokumenten verwendet. Da die Leserichtung horizontal ist, sind die Anwendungen speziell auf die Erkennung von horizontal angeordneten Zeichenketten trainiert. In natürlichen Bildern können ebenfalls horizontal ausgerichtete Zeichenketten vorkommen. Bei der OCR-Erkennung von horizontalen Texten bleibt die Reihenfolge der einzelnen Zeichen größtenteils beibehalten, somit erscheint eine Vertauschung der Zeichen als unnötig. Dennoch können einzelne Zeichen im erkannten Text entweder komplett fehlen oder falsch erkannt worden sein bzw. überflüssige Zeichen vorkommen. Am sinnvollsten für den Vergleich von horizontalen Texten in natürlichen Bildern erscheint somit die LCS- sowie die Levenshtein-Distanz.

Im Gegensatz zu eingescannten Dokumenten können in natürlichen Fotoaufnahmen auch einzelne Wörter vorkommen, bei denen die Zeichen vertikal oder einer Kurve bzw. einem Kreis folgend angeordnet sind. Da OCR-Anwendungen auf die Erkennung

horizontal angeordneter Texte trainiert sind, wird der Zusammenhang zwischen den einzelnen vertikal angeordneten Zeichen womöglich nicht erkannt, wodurch die Zeichen der vertikal angeordneten Zeichenkette in vermischter Reihenfolge erkannt werden können. Folglich spielt auch die Möglichkeit der Vertauschung von Zeichen eine größere Rolle, woraus sich ein Vorteil für die Damerau-Levenshtein- oder die Jaro-Distanz für den Vergleich zwischen annotiertem und erkanntem Text ergeben könnte.

Aufbauend auf den Definitionen der Distanzmaße bzw. des normalisierten Ähnlichkeitsmaßes für den Vergleich von Zeichenketten werden im folgenden Abschnitt aktuelle OCR-Anwendungen basierend auf dem NEOCR-Datensatz bzgl. der Texterkennung in natürlichen Bildern aus verschiedenen Blickwinkeln untersucht.

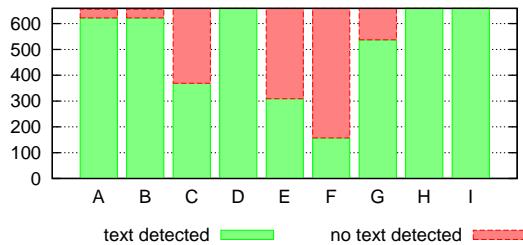
8.1.6 Evaluation aktueller OCR-Anwendungen

Um einen Gesamteindruck zu bekommen, wie gut aktuelle freie und kommerzielle OCR-Software Text in natürlichen Fotoaufnahmen erkennen, wurden verschiedene OCR-Anwendungen basierend auf dem NEOCR-Datensatz evaluiert.

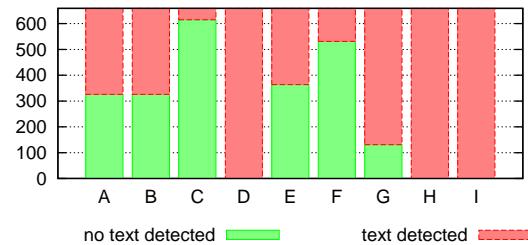
In einem ersten Schritt wurden anhand von 9 ausgewählten Testbildern aus dem NEOCR Datensatz diejenigen OCR-Anwendungen herausgesiebt, welche generell keine brauchbaren Ergebnisse geliefert haben. Dabei wurden verschiedene Konfigurationen für die Werkzeuge durchprobiert, da die Parameter der OCR-Anwendungen in erster Hinsicht auf eingescannte Dokumente optimiert sind. Die Namen der OCR-Anwendungen wurden anonymisiert, da nur ein Gesamteindruck der Texterkennung ermittelt werden sollte, ob OCR bereits für die Annotation von natürlichen Fotoaufnahmen einsetzbar ist. Für die ausgewählten OCR-Anwendungen wurde bei den nachfolgenden Tests diejenige Konfiguration verwendet, welche bei den 9 Testbildern die besten Ergebnisse geliefert hat. A und B stehen für die gleichen Anwendungen jedoch mit unterschiedlichen Vokabularen (A Englisch und B Deutsch). Sofern nicht getrennt angegeben, wurde bei den nachfolgenden Auswertungen für den Vergleich der erkannten Texte mit der annotierten Grundwahrheit als Texterkennungsrate die normalisierte Ähnlichkeit aus Gleichung 8.12 mit der Levenshtein-Distanz verwendet.

8.1.6.1 Textwahrnehmung in kompletten Bildern

Im ersten Szenario wurde die Wahrnehmung von Text in kompletten Bildern untersucht. Bilder, welche keinen Text enthalten, sollten von den OCR-Anwendungen abgewiesen werden, d. h. die Ausgabe der OCR-Anwendungen sollte leer sein oder eine entsprechende Fehlermeldung beinhalten. Ob der Text letztendlich richtig entziffert werden konnte, wurde bei diesem Test vorerst nicht berücksichtigt. Da im NEOCR-Datensatz alle Bilder Text beinhalteten, wurden zusätzlich Bilder ohne Text manuell aus dem MIR-Flickr-25000¹-Datensatz von [HL08] ausgewählt. Für den Test wurde der gesamte NEOCR-Datensatz und die gleiche Anzahl an textfreien Bildern (659 Bilder) herangezogen. Die Ergebnisse für die Textwahrnehmung in Textbildern sind in Abbildung 8.4(a), für textfreie Bilder in Abbildung 8.4(b) abgebildet.



(a) Bilder mit Textinhalt (aus NEOCR Datensatz)



(b) Bilder ohne Textinhalt (aus MIR Flickr-25000)

Bild 8.4: Textwahrnehmung durch ausgewählte OCR-Anwendungen in kompletten Bildern mit und ohne Textinhalt.

Abbildung 8.4(a) zeigt das Verhältnis von Bildern mit Text aus dem NEOCR-Datensatz, bei denen die OCR-Anwendungen Text gefunden haben (ohne zu beachten, ob die entdeckten Texte korrekt entziffert wurden). Der grüne Bereich stellt den Anteil der Bilder dar, bei denen Text gefunden wurde (richtig Positive), der rote Bereich steht für abgewiesene Bilder (falsch Positive).

Abbildung 8.4(b) zeigt das Verhältnis von Bildern ohne Text aus dem MIR-Flickr-25000-Datensatz, bei denen die OCR-Anwendungen Text gefunden haben. Der grüne Bereich stellt den Anteil an abgewiesenen Bildern dar, bei denen kein Text gefunden wurde (richtig Negative), der rote Bereich steht für Bilder, in denen fälschlicherweise Text erkannt wurde (falsch Negative). Die Genauigkeit bei der Trennung von Bildern mit und ohne Text ist bei den aktuellen OCR-Anwendungen noch recht gering. Die Anwendungen

¹ <http://press.liacs.nl/mirflickr/>

D, H und I haben immer Text in den Fotos gefunden, unabhängig davon, ob tatsächlich Text im Bild enthalten war.

8.1.6.2 Texterkennung in Bildausschnitten

Im Folgenden werden für die Evaluation der Texterkennung nur die annotierten Textrahmen als Bildausschnitte herangezogen. Bei den Auswertungen für einzelne Dimensionen ist jeweils die Anzahl der auf die entsprechende Einschränkung zutreffenden Bildausschnitte angegeben. Es werden sowohl die Ergebnisse für ggf. verzerrte seitenparallele Textrahmen als auch für entzerrte Texte angegeben. Die Entzerrung der Bildausschnitte wurde basierend auf den annotierten Verzerrungsparametern mittels ImageMagick gelöst. Abbildung 8.5 zeigt Beispiele aus dem NEOCR-Datensatz für ursprüngliche seitenparallele Textrahmen mit mehr oder weniger perspektivisch verzerrtem Textinhalt. In Abbildung 8.6 sind die zugehörigen entzerrten Bildausschnitte zu sehen. Probleme bereiten dabei Bilder mit geringer Auslösung, Texte, die nicht komplett im Ausschnitt enthalten sind, und gewölbte Texte. Allgemein sind die resultierenden entzerrten Bilder jedoch recht zufriedenstellend.

Gesamterkennungsrate

Bevor die OCR-Anwendungen nach verschiedenen Kriterien untersucht werden, wird zuerst die gesamte Erkennungsrate bei allen Bildausschnitten ermittelt. Bei den Bildausschnitten handelt es sich um die annotierten Textrahmen, welche die Textbereiche in Bildern durch seitenparallele Rechtecke umschließen. Dadurch muss der Text durch die OCR-Anwendungen nicht mehr erst im Bild gefunden, sondern nur noch richtig gedreht und entziffert werden. Es wird weiterhin untersucht, inwieweit sich die Entzerrung der Texte mittels des Verzerrungsvierecks auf die Erkennungsrate auswirkt.

Abbildung 8.7 zeigt die durchschnittliche normalisierte Ähnlichkeit unter Verwendung der Levenshtein-Distanz zwischen dem von OCR-Anwendungen erkannten und als Grundwahrheit annotierten Text. Bei jedem Verfahren wurde der beste Wert aus vier verschiedenen Versuchen ermittelt: original Farbbild, Graustufen sowie die jeweiligen negierten Versionen. Zusätzlich zum Maximalwert wird auch der Durchschnitt und der kleinste Ähnlichkeitswert der Versuche angegeben.



Bild 8.5: Beispiele für verzerrte Textausschnitte.



Bild 8.6: Beispiele für entzerrte Textausschnitte.

Anordnung der Zeichen

Eingescannte Dokumente enthalten fast ausschließlich horizontal angeordnete Zeichen, während bei Texten in natürlichen Fotoaufnahmen häufig auch vertikal und zirkular angeordnete Texte vorkommen. Um die Ergebnisse der Auswertung bezüglich diesem Merkmal nicht durch andere, möglicherweise sehr einflussreiche Eigenschaften zu vermischen, werden nur Bildausschnitte herangezogen, welche als nicht abgedeckt, nicht invertiert und nicht kompliziert (nicht *difficult*) eingestuft sind. Die Werte der durchschnittlichen Ähnlichkeit zwischen erkanntem und annotiertem Text sind in Abbildung 8.8 dargestellt. Man erkennt einen deutlichen Nachteil bei der Erkennung von vertikal und zirkular angeordneten Texten, wobei auch die Entzerrung hier wenig Abhilfe schafft.

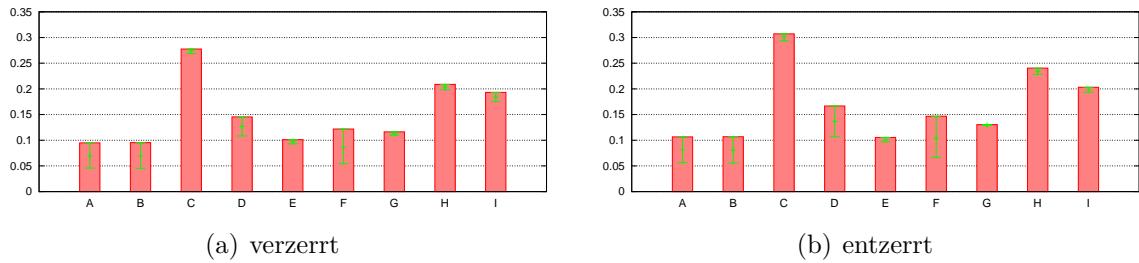


Bild 8.7: Texterkennungsrate für alle Textausschnitte (5238).

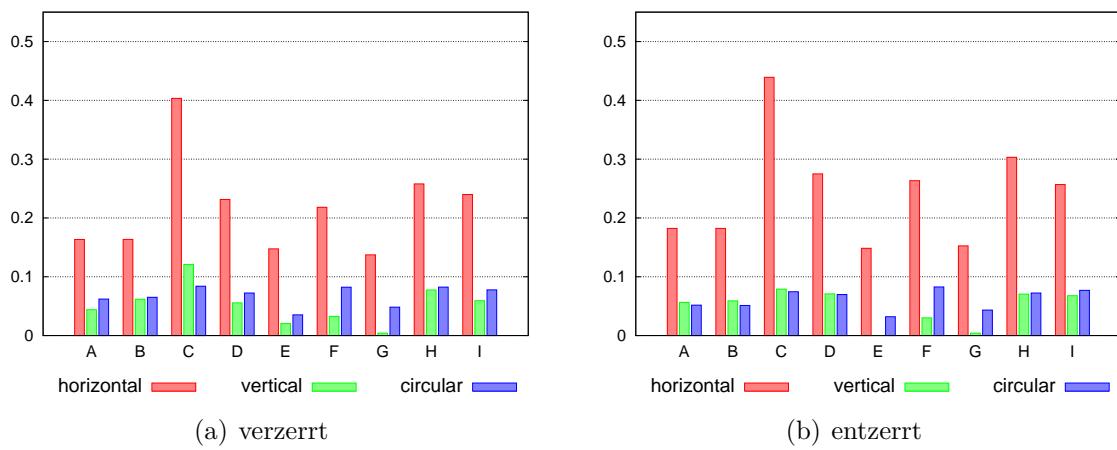


Bild 8.8: Texterkennungsrate für Textausschnitte nach Anordnung gruppiert: horizontal (*horizontal*, 2598), vertikal (*vertical*, 24) und zirkular (*circular*, 97).

Wie in Abschnitt 8.1.5.7 erläutert, könnte für die Evaluation der Erkennung von zirkularem und vertikalem Text der Einsatz von anderen Distanzfunktionen sinnvoll sein. Die Auswertungen mit unterschiedlichen Distanzfunktionen sind in den Abbildungen 8.9, 8.10 und 8.11 gegenübergestellt. Zwar wird bei allen Statistiken mit Jaro- und Jaro-Winkler-Distanz ein höherer Wert erreicht als mit der Levenshtein-Distanz, jedoch ist hierbei nicht von einem nennenswerten Vorteil der Vertauschungsmöglichkeit einzelner Zeichen auszugehen, da die höheren Werte auch bei horizontalem Text ersichtlich sind. Auch die Verhältnisse zwischen den verschiedenen Textanordnungen bleiben ähnlich, womit kein wesentlicher Vorteil für die Verwendung anderer Distanzmaße sichtbar ist.

Inversion

Ein weiterer wesentlicher Unterschied zu eingescannnten Dokumenten ist, dass Text in natürlichen Fotoaufnahmen oft auch hell auf dunklem Hintergrund erscheint. Um die

8. Verbesserung der Annotation durch Text in natürlichen Fotoaufnahmen

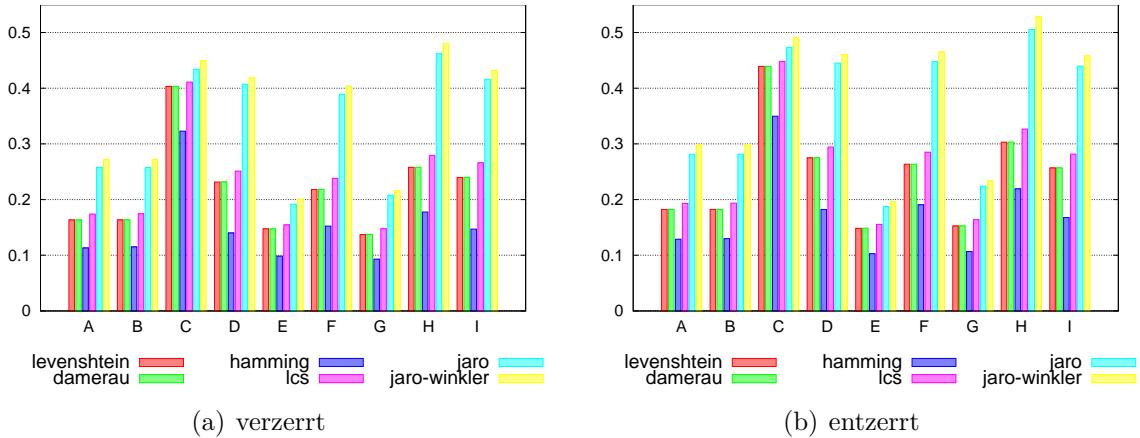


Bild 8.9: Vergleich verschiedener Distanzmaße für horizontale Ausschnitte (2598).

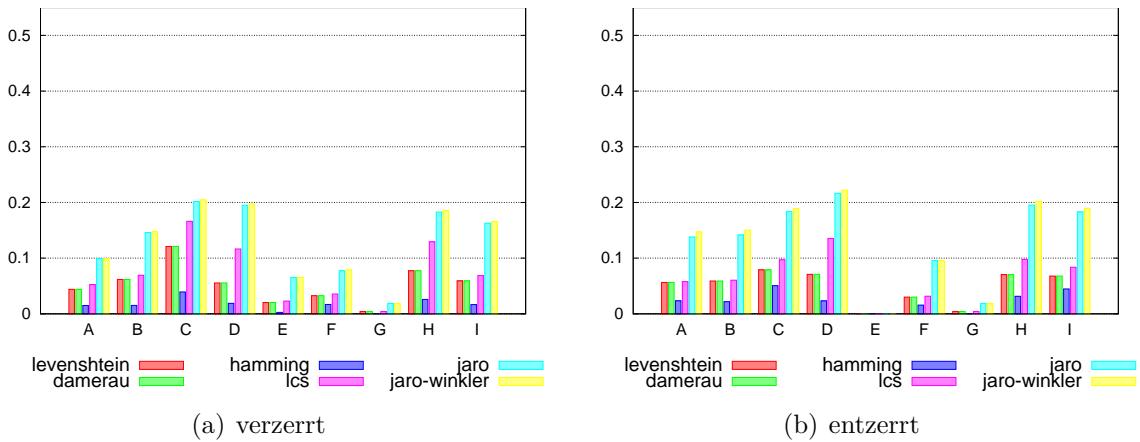


Bild 8.10: Vergleich verschiedener Distanzmaße für vertikale Ausschnitte (24).

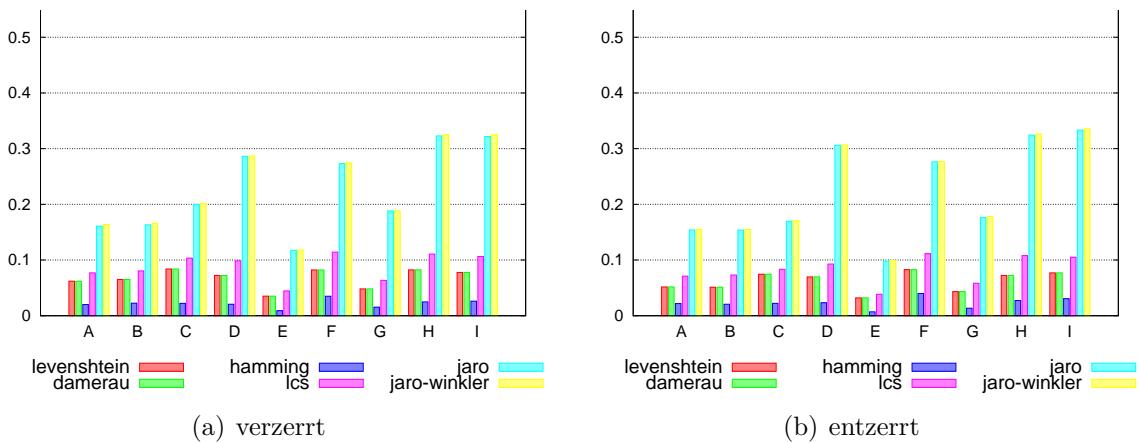


Bild 8.11: Vergleich verschiedener Distanzmaße für zirkuläre Textausschnitte (97).

Ergebnisse der Auswertung nicht durch andere Eigenschaften zu beeinflussen, werden nur horizontale und als nicht kompliziert eingestufte Bildausschnitte ohne Abdeckung berücksichtigt.

Abbildung 8.12 zeigt für einige OCR-Anwendungen enorme Schwierigkeiten bei invertierten Texten (heller Text auf dunklem Hintergrund). Deswegen wurde zusätzlich ein Test mit negierten invertierten Bildern durchgeführt, welcher bei allen OCR-Anwendungen zu besseren oder gleich guten Ergebnissen führte. Es ist deutlich erkennbar, dass dunkler Text auf hellem Hintergrund wesentlich genauer erkannt wird.

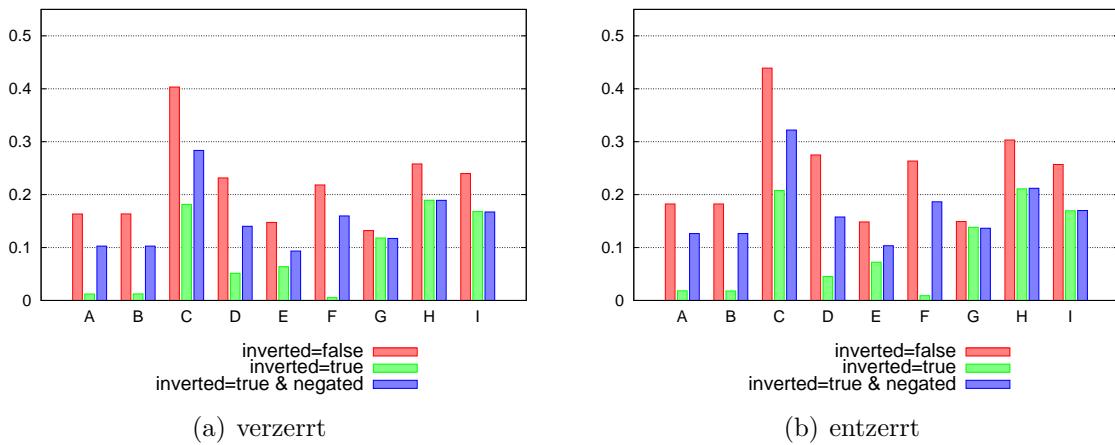


Bild 8.12: Texterkennungsrate für Textausschnitte nach Invertierung gruppiert: nicht invertiert ($\text{inverted}=\text{false}$, 2598), invertiert ($\text{inverted}=\text{true}$, 1641) und invertiert, jedoch vorab negiert ($\text{inverted}=\text{true} \& \text{negated}$, 1641).

Abdeckung

Abdeckungen können in natürlichen Fotoaufnahmen durch Objekte im Bild oder die Wahl des Ausschnitts durch den Fotografen entstehen. Je nachdem ob die Abdeckung horizontal oder vertikal ist, fehlen entweder Teile von allen Buchstaben oder nur einzelne Zeichen aus dem Text. Während das erste Problem schnell zur Nichterkennung des Textes führen kann, kann das zweite Problem bis zu einem gewissen Grad durch die Verwendung von Vokabularen entschärft werden.

In Abbildung 8.13 wird die Erkennungsrate zwischen abgedeckten und abdeckungsfreien (sowie gleichzeitig horizontalen, nicht komplizierten und nicht invertierten) Texten verglichen, wobei der Unterschied bei abgedeckten Texten deutlich zu erkennen ist. Die Erkennungsraten für abgedeckte Texte wurden in Abbildung 8.14 auf horizontale und vertikale Abdeckungen unterteilt, wobei die Vermutung sich bestätigte, dass vertikale

Abdeckungen (d. h. einzelne fehlende Zeichen) sich weniger negativ auf die Qualität der Texterkennung auswirken.

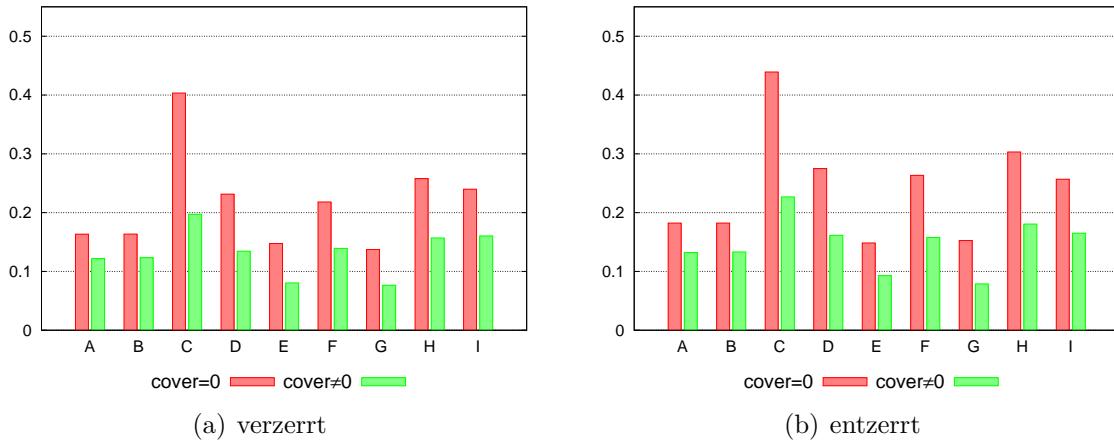


Bild 8.13: Texterkennungsrate für Textausschnitte nach Abdeckung gruppiert: nicht abgedeckt ($cover=0$, 2598), abgedeckt ($cover \neq 0$, 336).

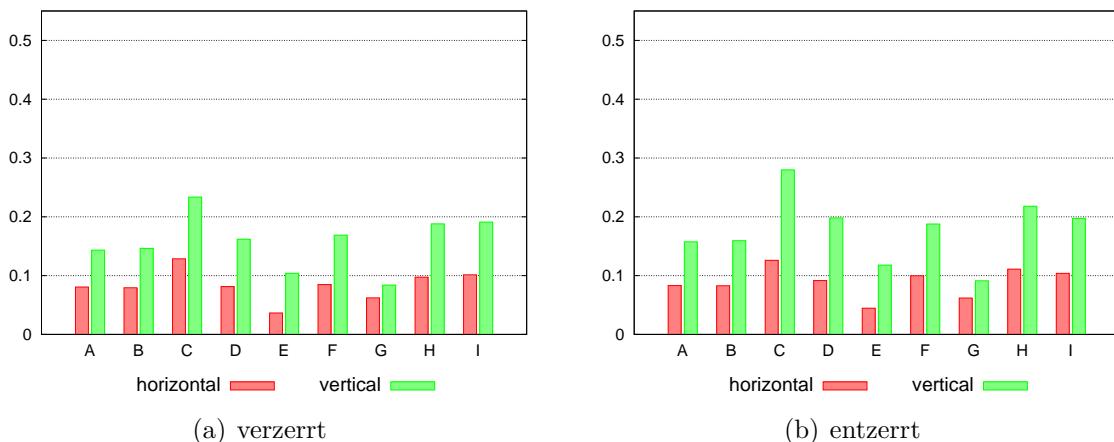


Bild 8.14: Texterkennungsrate für Ausschnitte nach Abdeckungsart (*orientation*) gruppiert: horizontal (*horizontal*, 115), vertikal (*vertical*, 221).

Abbildung 8.15 zeigt die vermutete Annahme, dass mit zunehmender Abdeckung die Erkennungsrate zurückgeht. Diese Annahme zeigt sich auch in Abbildung 8.16, wobei hier zusätzlich ersichtlich ist, dass einige Anwendungen einen Vorteil aus Vokabularen ziehen, da einzelne fehlerhafte Buchstaben leichter erkennbar und austauschbar sind. Deshalb ist die Qualität der Ergebnisse für sprachabhängige Wörter offensichtlich besser als bei sprachunabhängigem Text, wie z. B. Abkürzungen oder Ziffern.

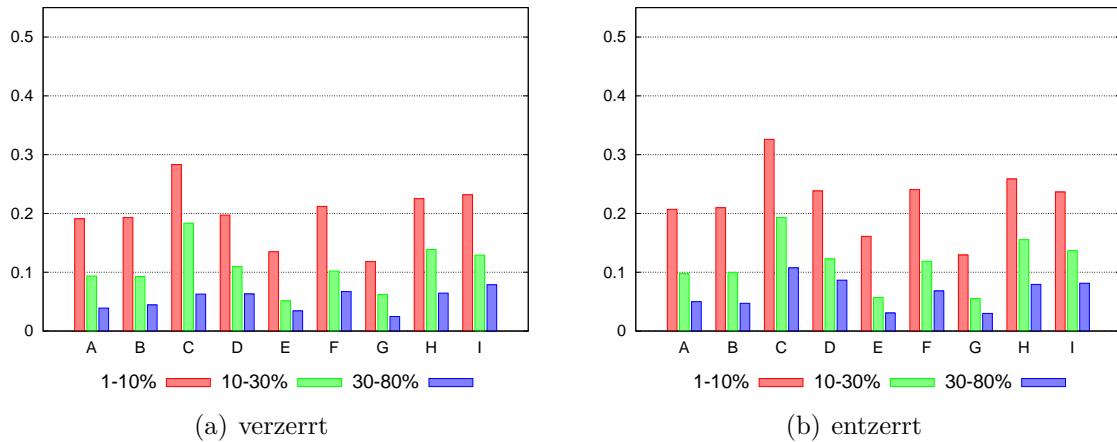


Bild 8.15: Texterkennungsrate für Ausschnitte nach prozentualer Abdeckung gruppiert: 0-10 % (135), 10-30 % (134), 30-80 % (67).

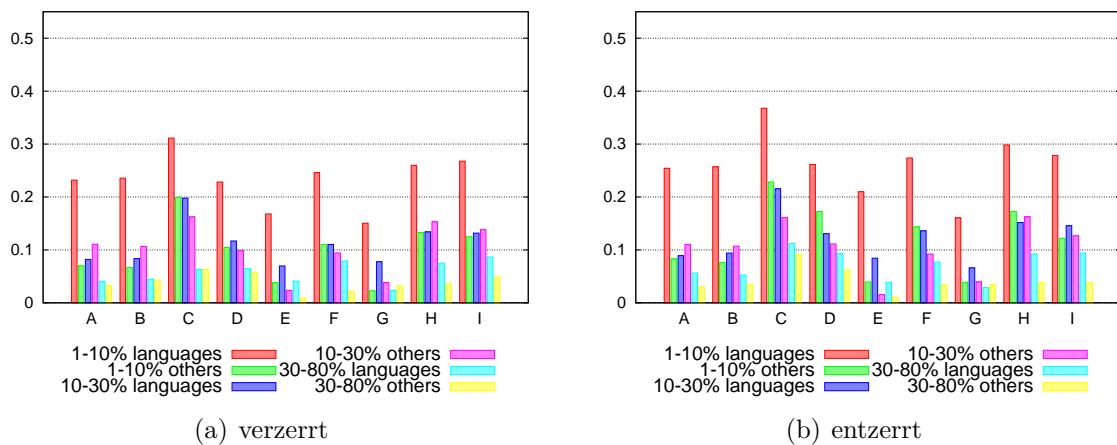


Bild 8.16: Texterkennungsrate für Ausschnitte nach prozentualer Abdeckung und Sprache gruppiert: 0-10 %: sprachabhängig (*languages*, 101), sprachunabhängig (*others*, 34); 10-30 %: sprachabhängig (*languages*, 81), sprachunabhängig (*others*, 53); 30-80 %: sprachabhängig (*languages*, 53), sprachunabhängig (*others*, 14).

Textur

Beim Vergleich der Erkennungsraten bzgl. der Textur in Abbildung 8.17 zeigt sich, dass bei steigender Komplexität die Qualität der Texterkennung stark abnimmt. Die Anzahl der Ausschnitte mit starker Textur (high) ist deswegen so niedrig, weil viele schon als schwierig (*difficult*) eingestuft wurden, bei der Auswertung jedoch nur horizontale nicht invertierte und nicht als schwierig eingestufte Texte ohne Abdeckung berücksichtigt wurden.

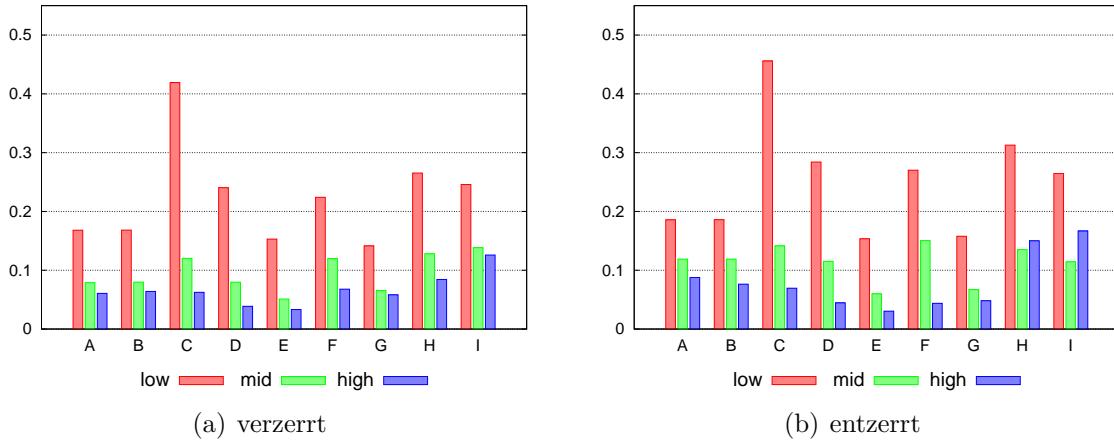


Bild 8.17: Texterkennungsrate für Ausschnitte nach Farbe bzw. Textur gruppiert: gering (*low*, 2463), mittel (*mid*, 123) und hoch (*high*, 12).

Helligkeit

Für die Evaluation der Texterkennung bzgl. der Helligkeit wurde der Wertebereich für Helligkeitsangaben auf 5 Gruppen aufgeteilt. Es wurden nur abdeckungsfreie horizontale nicht invertierte und nicht als schwierig eingestufte Ausschnitte berücksichtigt. Beste Erkennungsraten bei den Auswertungen in Abbildung 8.18 fallen dabei in die selben Helligkeitsbereiche, wo auch eingescannte Textdokumente zu finden sind.

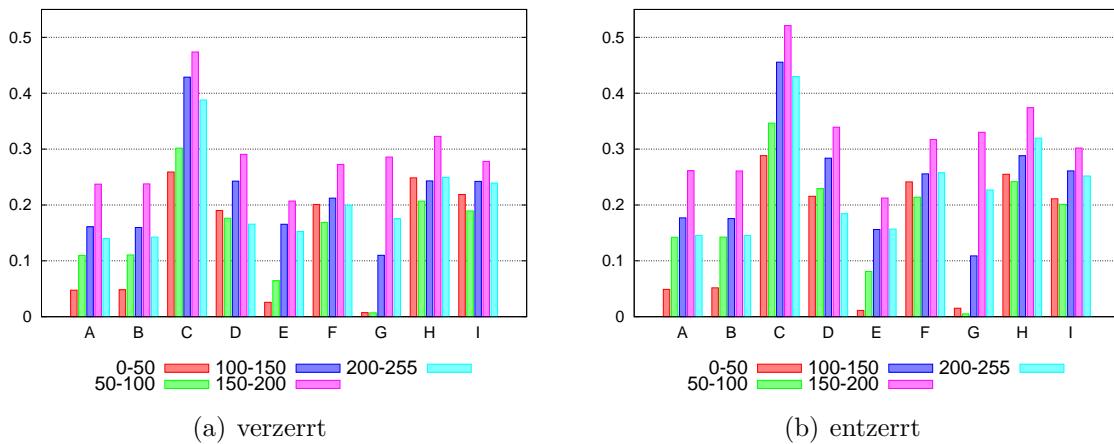


Bild 8.18: Texterkennungsrate für Textausschnitte nach Helligkeit gruppiert: 0-50 (135), 50-100 (494), 100-150 (1031), 150-200 (687) und 200-255 (251).

Kontrast

Der Wertebereich für Kontrast wurde basierend auf der Verteilung in Abbildung 8.3(b) auf 3 Gruppen aufgeteilt. Abbildung 8.19 zeigt die Erkennungsraten für die entsprechenden Kontrastbereiche, wobei offensichtlich ist, dass viele OCR-Anwendungen im Bereich 0-20 fehlschlagen. Die beste Texterkennung ist – ähnlich wie auch bei der Helligkeit – im Bereich der gescannten Textdokumente zu sehen.

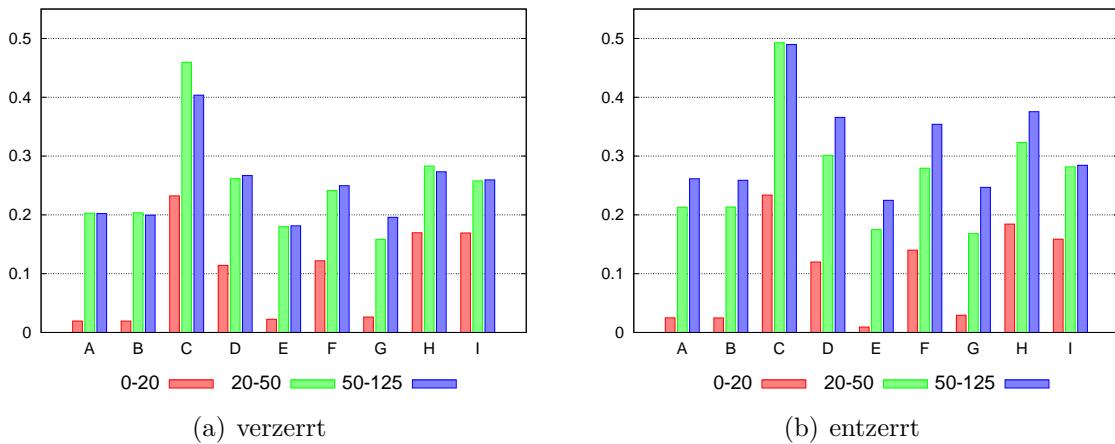


Bild 8.19: Texterkennungsrate für Textausschnitte nach Kontrast gruppiert: 0-20 (533), 20-50 (1625) und 50-125 (440).

Auflösung

Schlecht und sehr hoch aufgelöste Zeichen werden nach den Auswertungen in Abbildung 8.20 ungenauer erkannt. Das ist dadurch begründbar, dass bei eingescannten Dokumenten die eingenommene Fläche für ein Zeichen im Schnitt im Bereich von 2500 Pixeln liegt und sehr hohe Auflösungen auch nur durch wenige Scanner erreicht werden. Eine Ausnahme bildet die Anwendung C, welche mit schlecht aufgelösten Zeichen noch besser zurechtkommen scheint.

Rauschen

Rauschen wirkt sich ähnlich zur Textur mit steigender Intensität negativ auf die Erkennungsraten aus. Das bestätigt auch Abbildung 8.21, wobei die Anwendungen A, B, E und G mit stark verrauschten Bildern überhaupt nicht zurechtkommen.

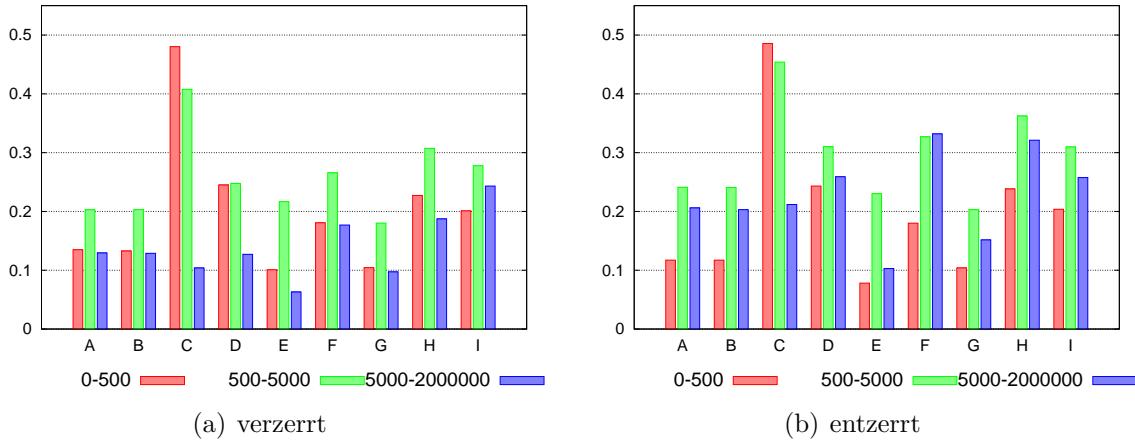


Bild 8.20: Texterkennungsrate für Textausschnitte nach Auflösung gruppiert: 0-500 (1134), 500-5000 (1157) und 5000-2000000 (307).

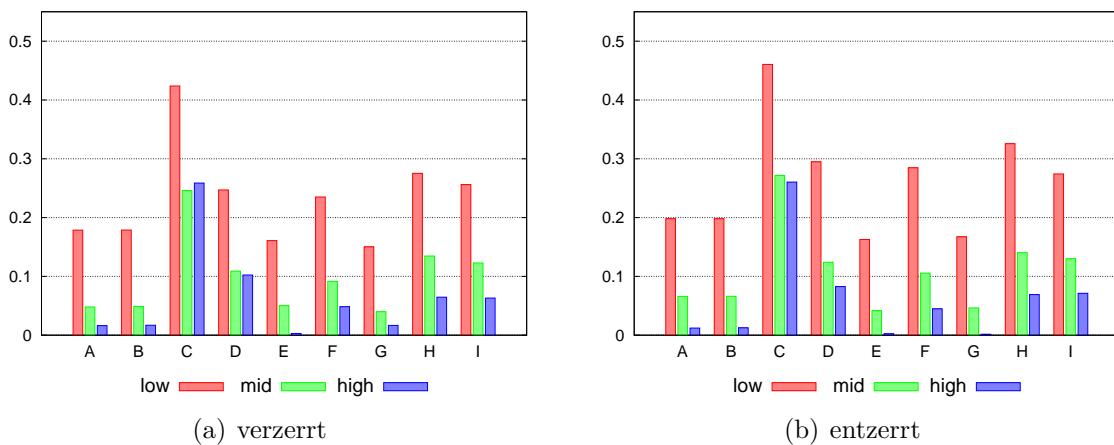


Bild 8.21: Texterkennungsrate für Textausschnitte nach Rauschen gruppiert: gering (*low*, 2307), mittel (*mid*, 241) und hoch (*high*, 50).

Unschärfe

Der Wertebereich für Schärfe wurde in 3 Gruppen unterteilt. Die Ergebnisse der Evaluation in Abbildung 8.22 zeigen (mit Ausnahme von Anwendung C) den Höchstwert bei Ausschnitten mit leichter Unschärfe. Dies deckt sich mit den Beobachtungen zur Auflösung, was den Zusammenhang zwischen der Auflösung und der definierten Messgröße für Schärfe unterstreicht.

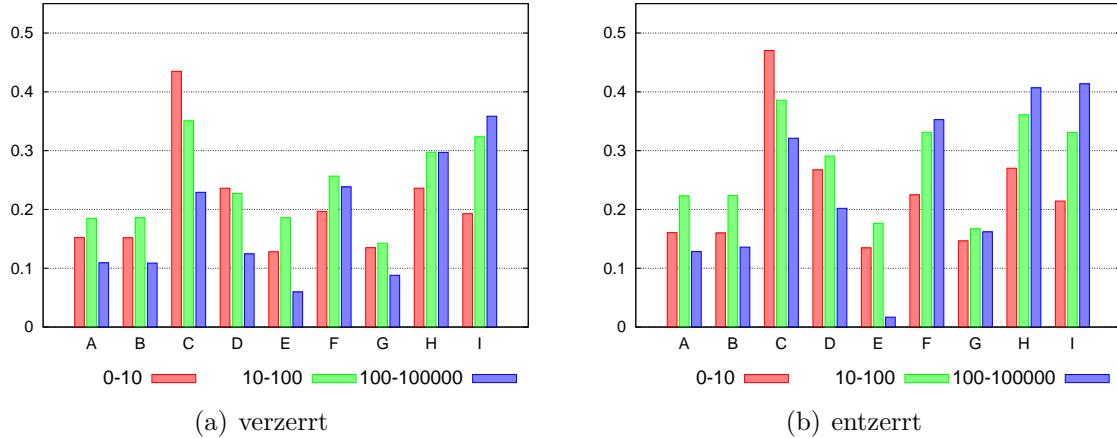


Bild 8.22: Texterkennungsrate für Textausschnitte nach Unschärfe gruppiert: 0-10 (1660), 10-100 (918) und 100-1000000 (20).

Verzerrung

Für die Bewertung der Auswirkung von Verzerrungen auf die Texterkennung wurden Bildausschnitte ohne perspektivische oder ausschließlich affine Verzerrungen mit perspektivischen Verzerrungen und Vielfachen von 90° verglichen. In Abbildung 8.23 zeigt sich deutlich, dass bei den meisten Anwendungen verdrehte Texte zu Problemen führen. Ebenfalls deutlich zu sehen ist die äußerst positive Auswirkung der Entzerrung mittels des annotierten Verzerrungsvierecks, was die Erkennungsrate vor allem bei affinen und perspektivischen Verzerrungen signifikant anhebt.

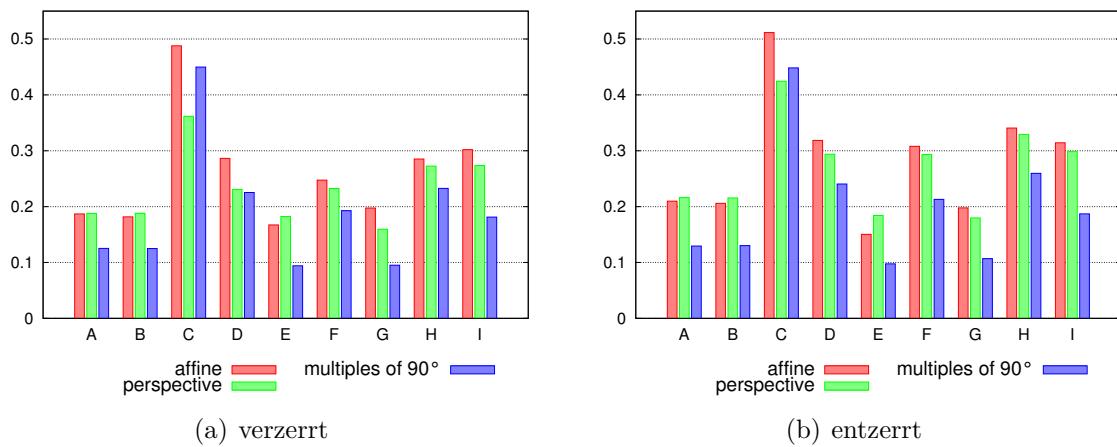


Bild 8.23: Texterkennungsrate für Textausschnitte nach Verzerrung gruppiert: keine perspektivische bzw. affine Verzerrung (*affine*, 164), andere Verzerrungsgrade (*perspective*, 1440) und Vielfache von 90° (*multiples of 90°*, 994).

Rotation

Bezüglich der Rotation wurden zwei verschiedene Szenarien untersucht.

In Abbildung 8.24 ist das Ergebnis für die Drehungen in Bereichen um das Vielfache von 90° zu sehen. Die meisten Anwendungen scheinen mit starken Rotationen nur schlecht umgehen zu können, bei manchen (z. B. A, B) ist jedoch zumindest eine eingebaute Verdrehung um 180° vermutbar.

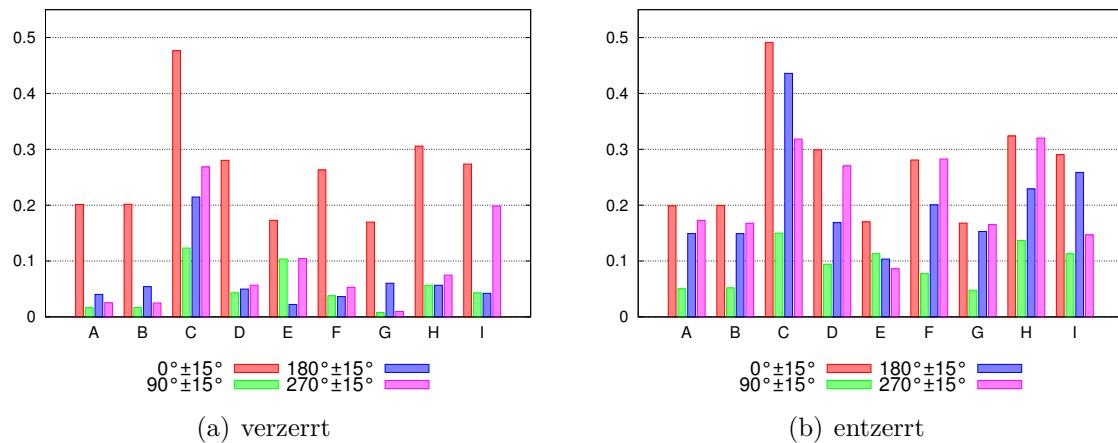


Bild 8.24: Texterkennungsrate für Textausschnitte nach Rotation um Vielfache von 90° gruppiert: $345^\circ - 15^\circ$ (2036), $75^\circ - 105^\circ$ (67), $165^\circ - 195^\circ$ (8) und $255^\circ - 285^\circ$ (227).

Im zweiten Szenario wurde untersucht, inwieweit die OCR-Anwendungen tolerant bzgl. leichten Verdrehungen um 0° sind. Abbildung 8.25 zeigt die besten Ergebnisse bei einer Abweichung um $\pm 5^\circ$, was darauf hindeutet, dass offenbar stets mit leichten Verdrehungen beim Einscannen von Dokumenten gerechnet wird.

Schriftart

Bei der Evaluation der Texterkennungsrate für verschiedene Schriftarten bestätigt sich in Abbildung 8.26 die Vermutung, dass die OCR-Anwendungen am besten mit Standardschriften zureckkommen. Für die Auswertung von Handschriften würde vermutlich der Einsatz einer speziell hierauf zugeschnittener Anwendung bessere Ergebnisse erzielen.

Sprache

Je nach dem ob OCR-Anwendungen sprachspezifische Vokabulare zur Texterkennung einsetzen, kann die Erkennungsrate erheblich gesteigert werden. Den ersten Hinweis für

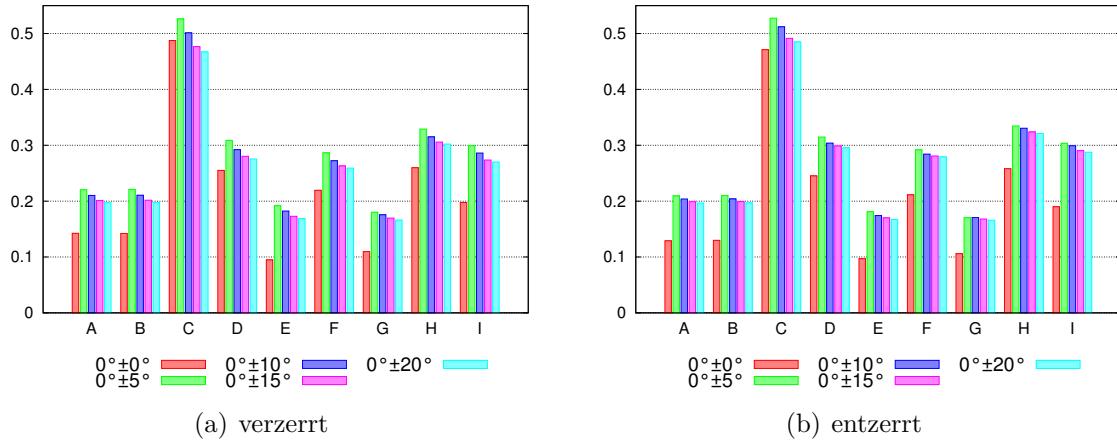


Bild 8.25: Texterkennungsrate für Ausschnitte nach Rotation um Winkelbereich bei 0° gruppiert: $0^\circ\pm0^\circ$ (861), $0^\circ\pm5^\circ$ (1681), $0^\circ\pm10^\circ$ (1898), $0^\circ\pm15^\circ$ (2036) und $0^\circ\pm20^\circ$ (2087).

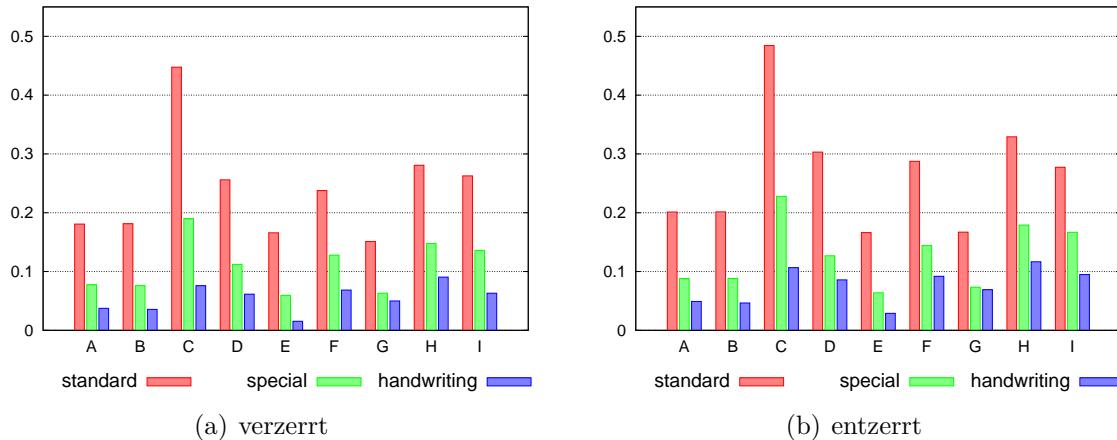


Bild 8.26: Texterkennungsrate für Textausschnitte nach Schriftarten gruppiert: Standard- (*standard*, 2215), Spezial- (*special*, 241) und Handschrift (*handwriting*, 142).

die Verwendung von Vokabularen liefert Abbildung 8.27, wo sprachabhängige Texte durchgängig bei jeder OCR-Anwendung bessere Erkennungsraten aufweisen.

Eine genauere Untersuchung für sprachabhängige Texte ist in Abbildung 8.28 zu sehen.

Bei sprachunabhängigen Texten ist in Abbildung 8.29 zu erkennen, dass alle Anwendungen Probleme mit der Erkennung von Zahlen und Abkürzungen haben. Wesentlich besser schneiden dabei Namen von Personen oder Firmen ab, was womöglich dadurch

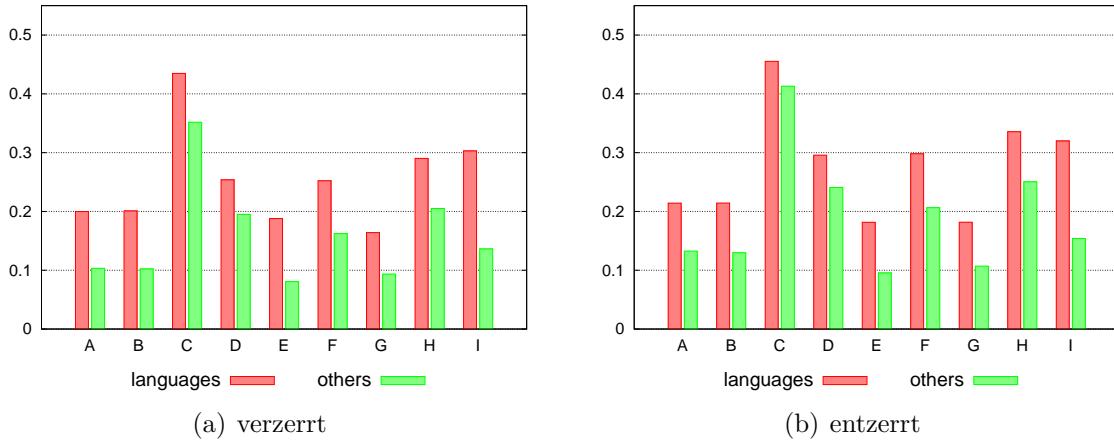


Bild 8.27: Texterkennungsrate für Textausschnitte nach Sprachabhängigkeit gruppiert: sprachabhängig (*languages*, 1616) und sprachunabhängig (*others*, 982).

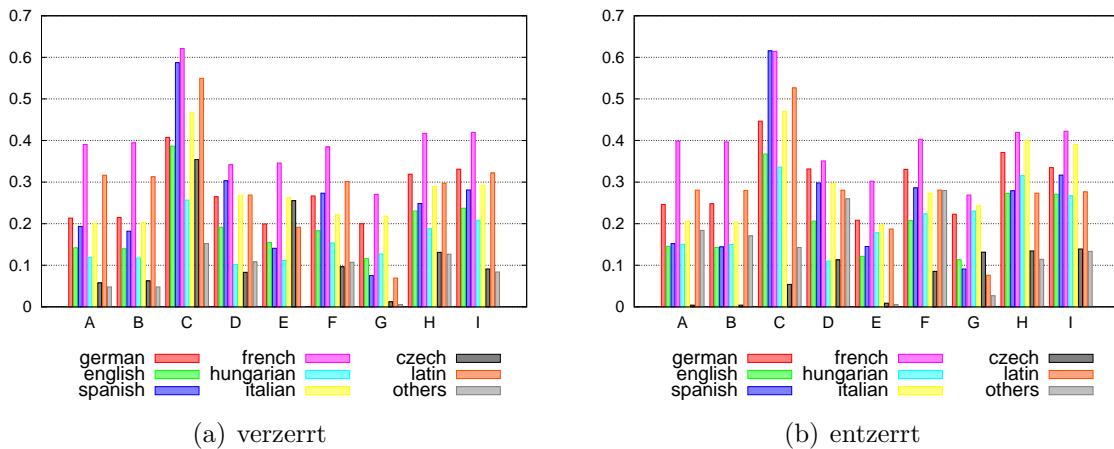


Bild 8.28: Texterkennungsrate für sprachabhängige Textausschnitte nach Sprachen gruppiert: deutsch (*german*, 908), englisch (*english*, 286), spanisch (*spanish*, 240), französisch (*french*, 81), ungarisch (*hungarian*, 30), italienisch (*italian*, 29), tschechisch (*czech*, 27), lateinisch (*latin*, 8) und andere (others: *turkish*, *portuguese*, *greek*, *swedish*, *russian*, *belgian*, 7).

bedingt ist, dass diese Wörter in den Vokabularen der Anwendungen zusätzlich enthalten sind.

8.1.7 Zusammenfassung

Die Ergebnisse der Evaluation in Abschnitt 8.1.6 liefern ein ernüchterndes Bild zum aktuellen Stand der Texterkennung in natürlichen Fotoaufnahmen. Ein wesentliches

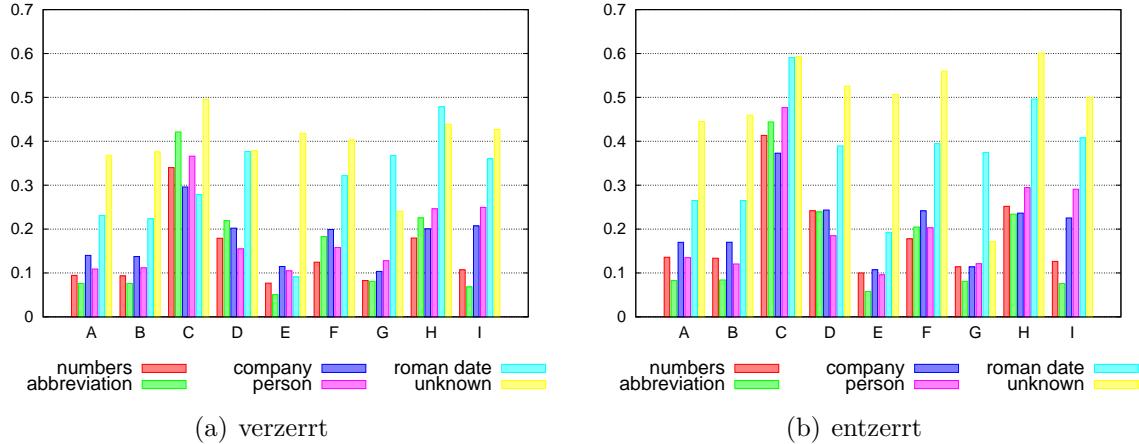


Bild 8.29: Texterkennungsrate für sprachunabhängige Textausschnitte nach Kategorien gruppiert: Zahlen (*numbers*, 431), Abkürzungen (*abbreviation*, 236), Firmen (*company*, 225), Personen (*person*, 72), römische Daten (*roman date*, 8) und unbekannt (*unknown*, 10).

Problem ist das Ausfiltern von Bildern ohne Textinhalt und die Lokalisierung von Textbereichen in Fotos. Viele der überprüften Anwendungen liefern – unabhängig davon, ob Text im Foto enthalten ist – bei jedem Bild erkannte Texte zurück.

Um das Problem der Textlokalisierung bei der Evaluation der Texterkennung selbst auszuschließen, wurden die Auswertungen auf den annotierten Bildausschnitten weitergeführt. Hier zeigte sich bezüglich der Genauigkeit der Texterkennung ebenfalls ein ernüchterndes Bild. Für alle Bildausschnitte des NEOCR-Datensatzes konnte lediglich eine Übereinstimmung um die 30% von erkanntem und annotiertem Text festgestellt werden.

Dank der Annotationen zu verschiedenen, für natürliche Fotoaufnahmen prägnanten Eigenschaften des neu erstellten NEOCR-Datensatzes konnten Problembereiche genau identifiziert werden. Zeichen, welche in vertikaler oder zirkularer Richtung angeordnet sind, werden schlechter erkannt als horizontale Texte. Ebenfalls bewahrheitet hat sich die Vermutung, dass die OCR-Anwendungen auf die Erkennung von hellem Text auf dunklem Hintergrund ausgerichtet sind. Interessant ist jedoch, dass invertierte Texte nach der Negation zwar durchweg von allen Anwendungen deutlich besser erkannt wurden, in der Erkennungsrate jedoch nicht an die der nicht invertierten Texte herankamen.

Die Auswirkungen von perspektivischen Verzerrungen, welche in natürlichen Fotoaufnahmen allgegenwärtig sind, konnte dank der annotierten Verzerrungsvierecke des NEOCR Datensatzes näher untersucht werden. Generell konnte eine Verbesserung der

Texterkennung um 2 bis 5% festgestellt werden. In speziellen Fällen, wie z. B. Rotationen um das Vielfache von 90°, war auch eine Verbesserung von bis zu 20% zu sehen.

In Tabelle 8.4 sind basierend auf den Auswertungen in Abschnitt 8.1.6 die optimalen Werte für die einzelnen Eigenschaften von natürlichen Fotoaufnahmen zusammengestellt. Die Werte sind weniger überraschend: Die meisten Optima liegen im Bereich von üblichen Werten für eingescannte Dokumente.

Aus der umfangreichen Evaluation von aktuellen OCR-Anwendungen wurde festgestellt, dass die Erkennung von Text in natürlichen Fotoaufnahmen leider nur sehr ernüchternd funktioniert. Wegen den erheblichen Mängel bei der Texterkennung kann OCR für die Verbesserung der Klassifikation und Annotation von natürlichen Fotoaufnahmen im aktuellen Stadium leider nur äußerst beschränkt eingesetzt werden.

EIGENSCHAFT	WERTEBEREICH	OPTIMUM
Textur	low, mid, high	low
Helligkeit	0 - 255	150 - 200
Kontrast	0 - 127	50 - 127
Invertierung	true, false	false
Auflösung	1 - 2 000 000	500 - 5 000
Rauschen	low, mid, high	low
Unschärfe	0 - 100 000	10 - 100
Verzerrung	sx: [-1;5], sy: [-1;1,5], tx: [0;1505], ty: [0;1419], px: [-0,03;0,07], py: [-0,02;0,02]	rx: [-15;22], ry: [-23;4], px: 0, py: 0
Rotation	0 - 360	0°±5°
Anordnung	horizontal, vertical, circular	horizontal
Abdeckung	0 - 100	0
Abdeckungsart	horizontal, vertical	vertical
Schriftart	standard, special, handwriting	standard
Sprache	german, english, spanish, hungarian, italian, latin, french, belgian, russian, turkish, greek, swedish, czech, portoguese, numbers, roman date, abbreviation, company, person, unknown	french
Schwierigkeit	true, false	false

Tabelle 8.4: Übersicht untersuchter Eigenschaften natürlicher Fotoaufnahmen mit jeweiligem Optimum.

8.2 Anreicherung der Annotation durch erkannten Text

Text in natürlichen Fotoaufnahmen kann zur genaueren Beschreibung des Bildinhalts beitragen. Zwar ist nach den Erkenntnissen aus Abschnitt 8.1 die Texterkennung in natürlichen Bildern zur Zeit noch nicht genau genug, um für die Verbesserung der Klassifikation eingesetzt zu werden, jedoch können erkannte Wörter nach einer entsprechenden Säuberung der Annotation hinzugefügt werden.

Als Ergänzung zur Annotation basierend auf erkannten Objekten wurde eine OCR-Anwendung in Pixtract integriert. Die Wahl fiel auf Tesseract OCR [Tes11], da es in der Evaluation in Abschnitt 8.1.6 unter den besten frei erhältlichen OCR-Anwendungen abschnitt und zusätzlich auch ein Vokabular bei der Erkennung einsetzt.

In Abschnitt 8.1.6 wurde gezeigt, dass die Erkennung von Text in natürlichen Fotoaufnahmen noch erhebliche Mängel aufweist. Auch Tesseract OCR erkennt viele chaotische Zeichenketten in natürlichen Bildern, die für eine Annotation unbrauchbar sind. Um die Qualität des erkannten Textes zu erhöhen, wird der von Tesseract OCR erkannte Text durch mehrere Stufen gefiltert.

Zuallererst werden alle Sonderzeichen entfernt, so dass nur alphanumerische Zeichen übrig bleiben. Anschließend werden die Zeichenketten in Kleinbuchstaben umgewandelt und nur Zeichenketten mit einer Mindestlänge von 3 Zeichen behalten. Dabei werden auch die Zeilenumbrüche entfernt. Nachfolgend werden die Wörter mittels WordNet und dem Natural Language Toolkit (NLTK) auf deren Grundform zurückgeführt. Die Lemmatisierung mittels WordNet dauert zwar länger als ein einfaches regelbasiertes Stemming (z. B. mittels dem Porter Stemmer), liefert jedoch grundsätzlich bessere Ergebnisse.

Da bei der Lemmatisierung mittels WordNet unbekannte Wörter beibehalten werden, müssen als letzter Schritt die Wörter in ihrer Grundform überprüft werden, ob sie in WordNet vorkommen. Am Ende bleiben diejenigen Wörter übrig, die WordNet bekannt sind, d. h. als Ergebnis werden nur sinnvolle Wörter ausgegeben. Die einzelnen Filter werden in Abbildung 8.30 anhand eines Beispiels aus ImageNet dargestellt.



Bild 8.30: Beispielbild aus dem ImageNet Datensatz für die Texterkennung und die einzelnen Filterungsschritte.

```

iiaw    W?
BIOTIN
300 ug
\ ~?2c\n?@k stravy %?
twb@?
\ /,Z~/ovy doplnok
@?Ymnovy
IOU
TABLET i
100 TABLIET

```

Listing 8.1: Der durch Tesseract
OCR erkannte Text
für Abbildung 8.30 im
Rohformat.

iiaw	
biotin	
300	
2cn?@k	
stravy	
twb@	
zovy	
doplnok	
ymnovy	
iou	
tablet	
i00	
tabliet	

Listing 8.2: Ergebnis nach Filterung
der Sonderzeichen
und Entfernung kurzer
Zeichenketten.

```
iaaw  
biotin  
300  
2cn?©k  
stravy  
twbÃ  
zovy  
doplnek  
ymnovy  
iou  
tablet  
i00  
tabliet
```

Listing 8.3: Ergebnis nach der Lemmatisierung.

```
biotin  
iou  
tablet
```

Listing 8.4: Endergebnis nach WordNet-Filterung.

Es wird angemerkt, dass die Texterkennung durch Tesseract OCR sehr stark von der Bildgröße und dem verwendeten Kompressionsverfahren abhängt. Das in Abbildung 8.30 dargestellte Bild ist ein JPEG und hat im ursprünglichen Format 663x1382 Pixel. Für dasselbe Bild in Größe 384x800 im JPEG-Format wird im Gegensatz zum Ergebnis in Listing 8.4 lediglich der Text „biotin“ ausgegeben, während im PNG-Format überhaupt kein Text gefunden wird. Optisch ist für das menschliche Auge kein Unterschied feststellbar, für die Texterkennungssoftware ist der Unterschied jedoch erheblich.

8.3 Zusammenfassung

In diesem Abschnitt wurden in natürlichen Fotoaufnahmen vorhandene Texte bzgl. der Verbesserung der Klassifikation und der Anreicherung der Annotation untersucht.

Dazu wurde in Abschnitt 8.1 festgestellt, dass die Texterkennung in Bildern durch aktuelle OCR-Anwendungen zur Zeit noch unzureichend ist. Eine Anreicherung der Annotation durch in Bildern erkannten Text wurde in Abschnitt 8.2 vorgestellt. Dabei wurde WordNet zur Säuberung der Ausgabe der verwendeten OCR-Anwendung eingesetzt.

Einen weiteren Einsatz findet WordNet bei der Verbesserung der Annotation direkt basierend auf den erkannten Objekten. Für die Liste der N wahrscheinlichsten erkannten Objekte wird zuerst untersucht, ob ggf. Oberbegriffe (Hyperonyme) einer ebenfalls erkannten Klasse in der Liste vorkommen. In einem Bild können zum Beispiel sowohl die

Kategorien „weißer Hai“, „Tigerhai“ und „Schwarzspitzenhai“ als auch die übergeordnete Kategorie „Hai“ erkannt werden. Mittels WordNet kann ermittelt werden, dass die ersten drei Kategorien zum Oberbegriff „Hai“ gehören; sie können somit unter letzterem Begriff subsumiert werden.

Falls es sich bei einem Objekt tatsächlich um einen „Hai“ handelt, bleibt diese Annotation vorhanden. Andernfalls können durch die Reduktion der Subklassen auf ihren Oberbegriff die richtigen Klassen weiter nach vorne rücken. Somit wird die Annotation insgesamt gesehen etwas unscharf, dabei jedoch besser.

Da für die Ermittlung der Oberbegriffe der gesamte Hypernym-Baum mittels WordNet aufgebaut wird, besteht somit auch die Möglichkeit die Annotation auf verschiedenen Ebenen zu erstellen. Je nach Wunsch des Benutzers können die übergeordneten Ebenen mit ausgegeben werden.

KAPITEL 9

VERGLEICH MIT ERWEITERBAREN ANSÄTZEN

In den vorherigen Abschnitten wurde eine erweiterbare Lösung für die objekterkennungsbasierte Annotation von Bildern vorgestellt und auf verschiedenen Datensätzen evaluiert. Der vorgestellte Ansatz wird in diesem Abschnitt mit anderen erweiterbaren Verfahren verglichen.

Für die Evaluation können nur wenige erweiterbare Ansätze herangezogen werden, da der Großteil der Verfahren auf Histogrammen von visuellen Wörtern aufsetzt, welche mittels eines allgemeinen visuellen Vokabulars erstellt werden. Bei der Erweiterung der Klassenmenge muss das visuelle Vokabular angepasst und somit auch die Histogramme sowie die Klassifikatoren komplett neu berechnet werden (vgl. Abschnitt 5.1.1). In dieser Evaluation werden somit Ansätze betrachtet, welche ohne ein allgemeines visuelles Vokabular arbeiten.

Viele erfolgreiche Ansätze führen die Klassifikation mittels SVMs durch, wobei diese Verfahren nicht erweiterbar sind (vgl. Kapitel 5). Mit einer zunehmenden Zahl von Klassen steigt bei der Verwendung von SVMs der Aufwand für das Hinzufügen einer neuen Klasse stetig an. In [DBLFF10] wird das Training von 1:N SVMs (vgl. Abschnitt 4.1.7.3) bei 10 000 Klassen auf 1 CPU-Jahr geschätzt. Da bei der Erweiterung der Klassenmenge um eine neue Klasse alle SVM-Klassifikatoren neu erlernt werden müssen, ist

diese Lösung nicht vertretbar. Des Weiteren wurde in [DBLFF10] auch gezeigt, dass NN-Klassifikatoren bei einer großen Anzahl von Klassen besser abschneiden als SVMs. Aus diesem Grund werden in dieser Evaluation ausschließlich Ansätze betrachtet, welche statt SVMs vorwiegend NN-Verfahren für die Klassifikation einsetzen.

Erweiterbare Lösungen, welche kein visuelles Vokabular verwenden, für die Klassifikation eine NN-Suche einsetzen und nach [BSI08] durchaus vergleichbare Resultate liefern wie die üblichen Ansätze mit Histogrammen von visuellen Wörtern und SVMs, sind verhältnismäßig rar. Für den Vergleich mit der in dieser Arbeit vorgestellten Lösung wurden die Verfahren aus [BSI08] und [AF10] implementiert und evaluiert (vgl. Abschnitt 4.1.6.2 und 9.1).

Im Nachfolgenden werden in Abschnitt 9.1 der Aufbau und die Einschränkungen der Evaluation beschrieben. Abschnitt 9.2 vergleicht die Ansätze bezüglich derer Klassifikationsgenauigkeit und Skalierbarkeit. In Abschnitt 9.3 werden die durchschnittlichen Antwortzeiten bei der Klassifikation von Bildern untersucht. Zuletzt wird in Abschnitt 9.4 die Erweiterbarkeit der Ansätze evaluiert. Die Ergebnisse der Vergleiche werden in Abschnitt 9.5 zusammengefasst.

9.1 Evaluationsaufbau

Zum Vergleich der Ansätze wird die durch Bounding Boxes annotierte Untermenge des ImageNet-Datensatzes mit 3 251 Klassen verwendet, da aktuell nur dieser Datensatz eine große Anzahl von manuell annotierten Klassen zur Verfügung stellt. Evaluationen mit mehreren 1 000 Klassen sind selten, was hauptsächlich durch den enormen Zeitaufwand für das Erlernen der Klassifikatoren mittels üblicher SVM-basierter Lösungen bedingt ist. Bislang ist auch [DBLFF10] die einzige Veröffentlichung, welche sich mit der Evaluation der Objekterkennung mit mehreren 1 000 Klassen befasst hat.

In der Evaluation wurde für jede der 3 251 Klassen eine disjunkte Menge von 20 Trainings- und 20 Testbildern zufällig bestimmt. Bei einigen Messungen wurden die Verfahren mit unterschiedlichen Anzahlen von Trainingsbildern evaluiert. Diese Abweichungen sind bei den entsprechenden Messungen gekennzeichnet.

Da die Verwendung verschiedener Bildgrößen die Klassifikation beeinflussen kann, wurden für die Vergleichbarkeit der Verfahren sowohl die Trainings- als auch die Testbilder auf eine einheitliche Größe skaliert. Alle Bilder wurden zuerst in den HSV-Farbraum

konvertiert, der V-Kanal (Helligkeitswerte) extrahiert und auf 128x128 Pixel normiert, wobei das Seitenverhältnis ignoriert wurde.

Für die Ermittlung der interessanten Punkte wurde der Hessian-Affine-Detektor sowie dichte Sampling mit einem Gitternetz mit Abständen von 8 Pixeln verwendet. Die ermittelten interessanten Punkte wurden durch SIFT-Deskriptoren beschrieben. Zur Extraktion und Beschreibung der Punkte wurde die Software aus [MS05] und [MTS⁺05] verwendet. Durch diese Schritte wird garantiert, dass jedes Verfahren auf der gleichen Datenbasis aufsetzt. Für Pixtract wurden zusätzliche Merkmale aus den Bildern extrahiert, welche in Kapitel 5 und 6 detailliert beschrieben wurden.

Zur Durchführung der Evaluation wurde ein Rechner mit einem 8-Kern Intel Xeon Prozessor (2 GHz, 64-Bit), 10 GB RAM und SuSE Linux Enterprise Server Version 11 verwendet. Auf eine Parallelisierung der Vergleiche mit einzelnen Klassen bzw. bei der Bearbeitung von mehreren Bildern wurde verzichtet, um eine zutreffendere Vergleichbarkeit zu ermöglichen. Parallele Berechnungen wurden lediglich auf der Ebene der Merkmale realisiert, indem für eine betrachtete Klasse und für ein gegebenes Bild die Vergleiche mit unterschiedlichen Merkmalen (z. B. GIST oder BoC) parallel erfolgten.

Im nachfolgenden Abschnitt werden die Einschränkungen der Evaluation für die jeweiligen betrachteten Verfahren erläutert.

Einschränkungen der Evaluation

Für die Evaluation wurden die Verfahren aus [BSI08] und [AF10] gemäß den Angaben in den entsprechenden Publikationen nachimplementiert. Dabei ergaben sich durch fehlende Informationen bzw. Abweichungen beim verwendeten Datensatz Einschränkungen bzgl. der Umsetzung der betrachteten Verfahren, auf welche in diesem Abschnitt kurz eingegangen wird. Im Wesentlichen handelt es sich dabei um Abweichungen bei der Größe der Bilder sowie den Einsatz von approximativen Verfahren zur schnelleren NN-Suche und damit der Klassifikation eines Bildes.

Einschränkungen bei [BSI08]

Für die Klassifikation der Bilder wurde der Naive-Bayes nächster Nachbar (NBNN) Klassifikator nach der Beschreibung in [BSI08] implementiert und verwendet. Hierfür wurden zuerst die SIFT-Deskriptoren (Merkmale) ${}^p\mathbf{c}$ für das zu klassifizierende Bild

(Muster) ${}^{\rho}\mathbf{f}(\mathbf{x})$ ermittelt und anschließend für jeden Deskriptor \mathbf{c}_i der nächste Nachbar $NN_{\Omega_\kappa}(\mathbf{c}_i)$ aus jeder Klasse $\Omega_\kappa \in \Omega$ bestimmt. Das Bild wurde derjenigen Klasse zugeordnet, bei dem die Summe der euklidischen Distanzen aller Deskriptoren des Bildes zu den jeweiligen nächsten Nachbarn der Klasse am kleinsten war. Bei der Berechnung der Distanzen zwischen den Deskriptoren wurde in [BSI08] zusätzlich die euklidische Distanz der Positionen der Deskriptoren im Bild ($l(\mathbf{c}_i)$) gewichtet durch den Parameter α berücksichtigt. Dadurch ergibt sich die folgende Formel für den *NBNN*-Klassifikator:

$$\Omega^* = \arg \min_{\Omega_\kappa} \sum_{i=1}^n \|\mathbf{c}_i - NN_{\Omega_\kappa}(\mathbf{c}_i)\|^2 + \alpha^2 \|l(\mathbf{c}_i) - l(NN_{\Omega_\kappa}(\mathbf{c}_i))\|^2. \quad (9.1)$$

[BSI08] nennt leider keinen konkreten Wert für α , lediglich, dass für alle Berechnungen der selbe Wert verwendet wurde. In Tests mit einer eingeschränkten Menge an Bildern bzw. Klassen erwies sich ein Wert von $\alpha = 0,02$ als geeignet, der auch in der nachfolgenden Evaluation eingesetzt wurde.

Zur Ermittlung der interessanten Punkte wurde in [BSI08] dense Sampling verwendet. Die Punkte wurden anschließend mittels SIFT-Deskriptoren beschrieben. Nach den Angaben in [BSI08] wurden die Bilder auf 5 verschiedene Größen skaliert. Leider wurden in [BSI08] weder die Bildgrößen noch die Abstände zwischen den interessanten Punkten für das dense Sampling erläutert. Des Weiteren fehlt auch die Angabe zur durchschnittlichen Anzahl der SIFT-Deskriptoren je Bild. Aus diesem Grund bzw. aus Gründen der besseren Vergleichbarkeit der Verfahren (vgl. Abschnitt 9.1) wurden in der nachfolgenden Evaluation Bilder mit 128 Pixel Seitenlänge sowie für das dense Sampling ein Gitternetz mit Abständen von 8 Pixeln verwendet.

Für die schnellere Abwicklung der Klassifikation der Bilder wurde in [BSI08] eine approximative NN-Suche nach [Mou10] eingesetzt. Approximative NN-Verfahren wurden in Abschnitt 5.2.2 kurz vorgestellt. Wesentlich ist die Angabe von einem ε , welches den Suchraum einschränkt und die Klassifikation schneller gestaltet, gleichzeitig jedoch auch einen Fehler einbringt. Leider wird in [BSI08] das Maß der Approximation nicht erwähnt. Aus diesem Grund wurde in der nachfolgenden Evaluation auf den Einsatz der approximativen NN-Suche verzichtet, was sich positiv auf die Klassifikation in Abschnitt 9.2, jedoch negativ auf die Antwortzeit in Abschnitt 9.3 auswirkt. Diese Konsequenzen werden in den entsprechenden Abschnitten im Detail diskutiert.

Einschränkungen bei [AF10]

In [AF10] wurden mehrere verschiedene NN-Klassifikatoren vorgeschlagen. Die Auswertungen in [AF10] haben den „Weighted Local Feature Distance Ratio Classifier“ (im Nachfolgenden als *LF Ratio* bezeichnet) als bestes befunden. Aus diesem Grund wurde in der nachfolgenden Evaluation lediglich der „NN Local Feature Classifier“ (*LF 1NN*) und der *LF-Ratio*-Klassifikator berücksichtigt.

Bei dem *LF-1NN*-Klassifikator wird jeder SIFT-Deskriptor \mathbf{c}_i eines Bildes ${}^{\rho}\mathbf{f}(\mathbf{x})$ derjenigen Klasse Ω_{κ} zugeordnet, aus dem sein nächster Nachbar stammt. Zusätzlich wird die euklidische Distanz $d(c_i, NN_{\Omega}(\mathbf{c}_i))$ zwischen zwei SIFT-Deskriptoren vermerkt. Das Bild ${}^{\rho}\mathbf{f}(\mathbf{x})$ wird anschließend derjenigen Klasse zugeordnet, bei der die aufsummierte Ähnlichkeit $1 - d(c_i, NN_{\Omega}(\mathbf{c}_i))$ über alle Deskriptoren \mathbf{c}_i des Bildes am höchsten ist. Der *LF-1NN*-Klassifikator wird somit folgendermaßen definiert:

$$\Omega^* = \arg \max_{\Omega_{\kappa}} \sum_{i=1}^n 1 - d(\mathbf{c}_i, NN_{\Omega}(\mathbf{c}_i)). \quad (9.2)$$

Der wesentliche Unterschied zum Verfahren in [BSI08] liegt darin, dass in [BSI08] für jeden Deskriptor \mathbf{c}_i in jeder einzelnen Klasse Ω_{κ} jeweils der nächste Nachbar bestimmt und in der Klassifikation berücksichtigt wird, während der *LF-1NN*-Klassifikator für jeden Deskriptor \mathbf{c}_i den nächsten Nachbarn aus der gesamten Klassenmenge Ω bestimmt und auch nur diese Distanzen bzw. Ähnlichkeiten bei der Klassifikation berücksichtigt.

Im Vergleich zum *LF-1NN*-Klassifikator bewertet der *LF-Ratio*-Klassifikator die Distanz zum nächsten Nachbarn in einer abgewandelten Form. Im Wesentlichen wird für jeden Deskriptor \mathbf{c}_i die Distanz zum nächsten Nachbarn aus allen Klassen Ω ($d(\mathbf{c}_i, NN_{\Omega}(\mathbf{c}_i))$) ins Verhältnis gesetzt mit der Distanz zum nächsten Nachbarn aus den Klassen $\Omega \setminus \Omega_{\kappa}$ ($d(\mathbf{c}_i, NN_{\Omega \setminus \Omega_{\kappa}}(\mathbf{c}_i))$). Der Deskriptor \mathbf{c}_i wird der Klasse Ω_{κ} mit dem nächsten Nachbarn zugeordnet. Die Distanz berechnet sich wie folgt:

$$\dot{\sigma}(\mathbf{c}_i) = \frac{d(\mathbf{c}_i, NN_{\Omega \setminus \Omega_{\kappa}}(\mathbf{c}_i))}{d(\mathbf{c}_i, NN_{\Omega}(\mathbf{c}_i))}. \quad (9.3)$$

Das Bild ${}^\rho \mathbf{f}(\mathbf{x})$ wird anschließend derjenigen Klasse zugeordnet, bei der die aufsummierte Ähnlichkeit $(1 - \dot{\sigma}(\mathbf{c}_i))^2$ über alle Deskriptoren \mathbf{c}_i des Bildes am höchsten ist. Somit wird der *LF-Ratio*-Klassifikator folgendermaßen definiert:

$$\Omega^* = \arg \max_{\Omega_\kappa} \sum_{i=1}^n (1 - \dot{\sigma}(\mathbf{c}_i))^2. \quad (9.4)$$

In [AF10] wurde ein eigens konstruierter Datensatz mit Fotos von 12 Wahrzeichen erstellt. Für die Ermittlung der interessanten Punkte wurden Bilder mit einer Seitenlänge von 500 Pixel verwendet. Im Vergleich zu den Bildern des ImageNet-Datensatzes waren die 1 227 Aufnahmen in [AF10] wesentlich größer. Die Anzahl der Trainingsbilder mit durchschnittlich 20 Bildern je Klasse war jedoch vergleichbar. Da nur sehr wenige Bilder des ImageNet-Datensatzes eine Seitenlänge von 500 Pixel erreichen und dadurch die Anzahl der Klassen stark eingeschränkt werden würde, wurden in der nachfolgenden Evaluation Bilder mit einer Seitenlänge von 128 Pixel verwendet.

Einschränkungen bei Pixtract

Für die Evaluation von Pixtract wurden ebenfalls Bilder mit einer Seitenlänge von 128 Pixel verwendet. Des Weiteren wurde die Einschränkung des Suchraums aus Abschnitt 5.2.2.2 bzw. 6.5 nicht berücksichtigt, da sowohl bei [BSI08] als auch bei [AF10] keine approximative NN-Suche in der nachfolgenden Evaluation eingesetzt wurde.

9.2 Klassifikation und Skalierbarkeit

Die Genauigkeit der Klassifikation der evaluierten Verfahren wird anhand der MAP für unterschiedliche Anzahlen von Klassen festgestellt. Da die jeweilige Kombination der Klassen die Erkennung stark beeinflusst, wird der Durchschnitt der MAP von mehreren Klassengruppen zur Bewertung herangezogen.¹ Für einen Testdurchlauf mit k Klassen wurden die 3 251 Klassen auf $3 251/k$ disjunkte Gruppen aufgeteilt, für jede Gruppe die MAP einzeln ermittelt und zuletzt der Durchschnitt über alle Gruppen gebildet. Somit kann der Einfluss der Zusammensetzung einer Gruppe von k Klassen abgeschwächt und

¹ Mit einer Gruppe von Klassen, deren Bilder sich stark unterscheiden, kann in der Regel genauer klassifiziert werden als mit einer Gruppe von Klassen, welche ähnliche Objekte abbilden.

ein allgemeiner Eindruck von der Erkennungsrate des jeweiligen Verfahrens gewonnen werden.

Abbildung 9.1 zeigt die MAP für unterschiedliche Anzahlen von Klassen für [BSI08], [AF10] und Pixtract. Bei [BSI08] wurde sowohl die Verwendung von dense Sampling (*NBNN Dense*) als auch die Ermittlung von interessanten Punkten mittels des Hessian-Affine Detektors (*NBNN HesAff*) evaluiert. Die Ergebnisse für die zwei verschiedenen Verfahren aus [AF10] sind in Abbildung 9.1 unter den Abkürzungen *LF 1NN* und *LF Ratio* zu sehen.

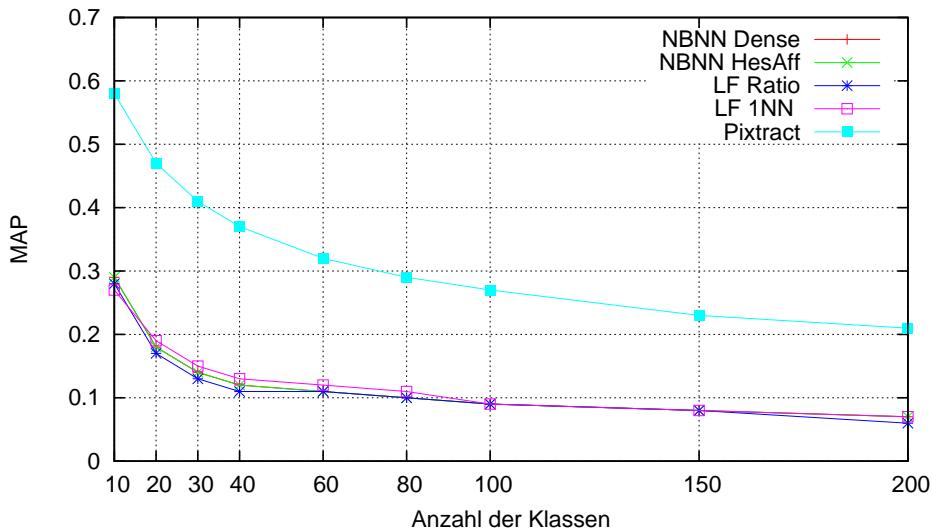


Bild 9.1: Vergleich der MAP für unterschiedliche Anzahlen von Klassen in Pixtract, sowie mit den Verfahren aus [BSI08] und [AF10]. *NBNN Dense* und *NBNN HesAff* repräsentieren den Ansatz aus [BSI08], wobei für *NBNN Dense* dense Sampling und für *NBNN HesAff* der Hessian-Affine Detektor verwendet wurde. Die beiden Verfahren aus [AF10] sind unter *LF 1NN* und *LF Ratio* dargestellt.

Zwischen den Ansätzen aus [BSI08] und [AF10] sind kaum Unterschiede feststellbar. Offensichtlich ist bei der Verwendung einer identischen Datenbasis keiner der in Abschnitt 9.1 vorgestellten NN-Klassifikatoren deutlich besser oder schlechter bzgl. der Genauigkeit der Objekterkennung. Auch die Einbindung der Positionen der Deskriptoren im Bild in die Distanzberechnungen der *NBNN*-Ansätze ergab keine nennenswerte Vorteile.

Bereits schon bei wenigen Klassen zeigte sich die Überlegenheit von Pixtract. Die Entwicklung der MAP bei der Hinzunahme von weiteren Klassen veranschaulicht eine deutlich bessere Skalierbarkeit von Pixtract im Vergleich zu den Verfahren aus [BSI08] und

[AF10]. Die Differenz ist daraus ableitbar, dass Pixtract mehrere Merkmale verwendet, die in ihrer Kombination zu einer besseren Erkennung führen. Des Weiteren konnte wegen fehlender Informationen der Ansatz aus [BSI08] nicht vollständig umgesetzt werden (siehe Abschnitt 9.1), was zur schlechteren Bewertung der *NBNN*-Methoden führte. Die mäßigen MAP-Werte für [AF10] können dadurch erklärt werden, dass die *LF-1NN*- und *LF-Ratio*-Klassifikatoren bislang nur auf sehr kleinen Datensätzen mit wesentlich größeren Bildern evaluiert wurden.

9.3 Antwortzeit

Verschiedene Verfahren zur Objekterkennung unterscheiden sich erheblich bzgl. der Antwortzeit bei der Klassifikation eines Bildes (vgl. Abschnitt 2.1.3). Diese Zeit ist im wesentlichen abhängig von der Anzahl der Merkmale, mit denen das neue Bild verglichen werden muss. Wesentliche Einflussfaktoren dabei sind:

- die Größe und Anzahl der Trainingsbilder,
- die Anzahl der Klassen insgesamt,
- die Art und Konfiguration der eingesetzten Detektoren,
- Reduktionsmaßnahmen für die Ermittlung der Klassenbeschreibungen sowie
- approximative Methoden zur Verringerung der Anzahl der Vergleiche und somit der Verkürzung der Suche.

Letztere werden, wie in Abschnitt 9.1 erläutert, in dieser Evaluation nicht betrachtet. Des Weiteren wurden die Vergleiche mit den einzelnen Klassen auch nicht parallel ausgeführt.

Für die Bewertung der Antwortzeit bei der Klassifikation eines Bildes wurden 100 Bilder zufällig ausgewählt. Anschließend wurden diese Bilder mit einer unterschiedlichen Anzahl von Klassen klassifiziert. Die Klassifikation wurde 10-fach wiederholt und nachfolgend die durchschnittliche Antwortzeit für die Klassifikation eines einzelnen Bildes berechnet. Die Größe der Bilder bzw. die Ermittlung der Merkmale entspricht der Beschreibung in Abschnitt 9.1.

Abbildung 9.2 zeigt die durchschnittlichen Antwortzeiten für die Klassifikation eines einzelnen Bildes unter Verwendung der untersuchten erweiterbaren Verfahren.

Es ist ein deutlicher Unterschied zwischen den durchschnittlichen Antwortzeiten der untersuchten Verfahren feststellbar.

In der Veröffentlichung [AF10] selber sind keine Angaben zu Antwortzeiten aufgeführt. Die unterschiedlichen Antwortzeiten der zwei Verfahren aus [AF10] in Abbildung 9.2 sind darauf zurückführbar, dass *LF 1NN* lediglich eine Suche nach dem nächsten Nachbarn in der Klassenmenge Ω durchführt, während *LF Ratio* die Suche nach dem nächsten Nachbarn im Anschluss zwar auf einer kleineren Klassenmenge $\Omega \setminus \Omega_\kappa$, aber dennoch ein zweites Mal unternehmen muss. Dadurch ergibt sich auch grob eine doppelte Antwortzeit für den *LF Ratio* Klassifikator.

In [BSI08] wurde (unter Verwendung der approximativen NN-Suche) die Zeit für die Klassifikation eines Bildes je Klasse mit durchschnittlich 1,6 Sekunden angegeben. Bei einem sequentiell ausgeführten Vergleich mit allen Klassen würde sich demnach auf 3 251 Klassen gesehen eine Antwortzeit von ca. 5 200 Sekunden ergeben. In Abbildung 9.2(b) wurde die durchschnittliche Antwortzeit bei 3 251 Klassen auf ca. 3 100 Sekunden hochgerechnet. Die Differenz zwischen den durchschnittlichen Antwortzeiten kann auf die Verwendung unterschiedlicher – jedoch leider nicht dokumentierter – Bildgrößen in [BSI08] zurückgeführt werden (vgl. Abschnitt 9.1).

Pixtract bietet durch seine kompakteren Merkmale im Vergleich zu [BSI08] und [AF10] eine wesentlich bessere durchschnittliche Antwortzeit bei der Klassifikation eines Bildes. Da die Größe der Bilder, die Anzahl der Trainingsbilder je Klasse sowie die Anzahl der Klassen für alle Verfahren identisch sind, liegt der wesentliche Grund für diese Differenz darin, dass [BSI08] und [AF10] die NN-Suche direkt auf den ermittelten SIFT-Deskriptoren durchführen. Im Vergleich dazu sind in Pixtract die einzelnen Klassen durch verschiedene Durchschnittswerte der Deskriptoren repräsentiert, was die Anzahl der Vergleiche bei der Klassifikation erheblich reduziert. Dieser Unterschied wird umso mehr verstärkt, je höher die Anzahl der Klassen ist.

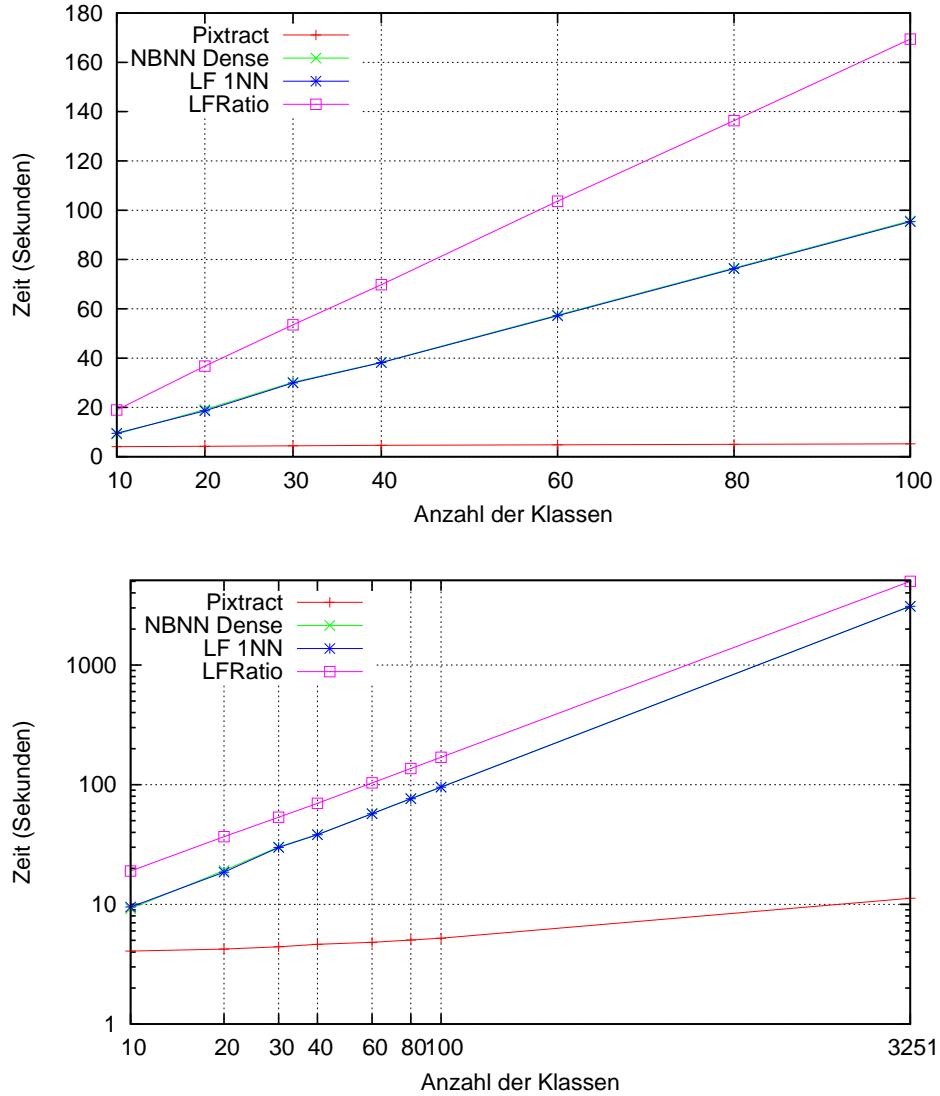


Bild 9.2: Vergleich der durchschnittlichen Antwortzeit für die Klassifikation eines Bildes für unterschiedliche Anzahlen von Klassen in Pixtract sowie mit den Verfahren aus [BSI08] und [AF10]. Da sich die Antwortzeiten deutlich unterscheiden, ist die gleiche Messung in Abbildung 9.2(b) mit einer logarithmischen Zeitachse dargestellt. Diese Abbildung enthält auch eine Hochrechnung (bzw. bei Pixtract eine tatsächliche Messung) der durchschnittlichen Antwortzeit für die Klassifikation eines Bildes bei der Verwendung von 3 251 Klassen.

9.4 Erweiterbarkeit

Erweiterbarkeit war ein primäres Ziel der hier vorgestellten Arbeiten. Gemäß der Definition in Abschnitt 2.3 wird unter Erweiterbarkeit die nachträgliche Aufnahme und Integration von neuen Klassen in das System ohne signifikantem Zeit- und Berechnungsaufwand verstanden. Wesentliche Faktoren beim Hinzufügen einer neuen Klasse bzgl. der Erweiterbarkeit sind:

- die Anzahl der Trainingsbilder,
- die Extraktion der Merkmale und die Berechnung der ObjectFPs bzw. der Histogramme von visuellen Wörtern sowie
- je nach Verfahren das Erlernen der Klassifikatoren.

Für den Vergleich der Erweiterbarkeit von Pixtract mit den Verfahren aus [BSI08] und [AF10] wurde die durchschnittlich benötigte Zeit sowie der durchschnittlich benötigte Speicherplatz für das Hinzufügen einer neuen Klasse in Abhängigkeit von der Anzahl der Trainingsbilder herangezogen.

Für die Messung der benötigten Zeit wurde die Zeitspanne zwischen der Entgegennahme der Bilder in ihrem Rohformat und der Aufnahme der neuen Klasse ins jeweilige Objektverzeichnis erfasst. Für die Evaluation wurden 100 zufällig Klassen mit jeweils 5, 10, 15 und 20 Testbildern zufällig ausgewählt und anschließend die durchschnittlich benötigte Zeit berechnet. Abbildung 9.3 zeigt die durchschnittlich erforderliche Zeit für die Aufnahme einer neuen Klasse bei unterschiedlichen Anzahlen von Trainingsbildern. Berücksichtigt wurde dabei im Wesentlichen die Zeit für die Extraktion der Merkmale und die Berechnung der ObjectFPs, da bei den betrachteten Ansätzen keine Klassifikatoren erlernt werden müssen.

Die großen Unterschiede bzgl. der benötigten Zeit für das Hinzufügen von neuen Klassen beruhen darauf, dass sowohl [BSI08] als auch [AF10] direkt auf den SIFT-Deskriptoren der einzelnen Trainingsbilder aufsetzen. Deswegen wurden die beiden Ansätze in Abbildung 9.3 auch zusammengefasst und lediglich nach der Art der Detektoren unterschieden. Im Gegensatz dazu werden in Pixtract basierend auf den SIFT-Deskriptoren klassenspezifische Vokabulare berechnet, welche eine Form von Durchschnittsbildern für die jeweiligen Klassen repräsentieren. Dieser zusätzliche Schritt ist verantwortlich für die Zeitdifferenz bei der Aufnahme einer neuen Klasse, führt jedoch später bei der Klassifikation von

9. Vergleich mit erweiterbaren Ansätzen

neuen Bildern zur erheblichen Zeitersparnis – vor allem bei einer hohen Gesamtanzahl an Klassen.

Die Aufnahme einer neuen Klasse in das Objektverzeichnis erfordert in Pixtract bei der Verwendung von 20 Trainingsbildern je Klasse im Schnitt ca. 3 Minuten. Diese Zeit ist durchaus vertretbar, vor allem wenn man berücksichtigt, dass sie unabhängig von der Größe des Objektverzeichnisses ist.

Für den Vergleich des durchschnittlich je Klasse benötigten Speicherplatzes bei den betrachteten Verfahren wurden die auf der Festplatte belegte Größe der abgespeicherten Merkmale sowie der zusätzlich benötigten Hilfsstrukturen herangezogen. In der Evaluation wurde der Mittelwert über alle 3 251 Klassen berechnet, wobei für jede Klasse 20 zufällig ausgewählte Trainingsbilder verwendet wurden. Abbildung 9.4 veranschaulicht die Unterschiede bzgl. des durchschnittlich belegten Speicherplatzes je Klasse.

Auch für diese Evaluation wurden die Verfahren [BSI08] und [AF10] zusammengefasst und nur nach der Art der Detektoren unterschieden, da beide Verfahren direkt auf den SIFT-Deskriptoren der Trainingsbilder arbeiten. Letzteres begründet auch den erheblichen Unterschied bzgl. des durchschnittlichen Speicherplatzbedarfs im Vergleich zu Pixtract. Im Gegensatz zu den SIFT-Deskriptoren je Bild bei [BSI08] und [AF10], werden in Pixtract je Klasse „aggregierte“ Merkmale abgespeichert. Dieses Vorgehen bei Pixtract ermöglicht neben einer erheblichen Zeitersparnis während der Klassifikation auch gleichzeitig eine Reduktion bzgl. des Speicherplatzbedarfs.

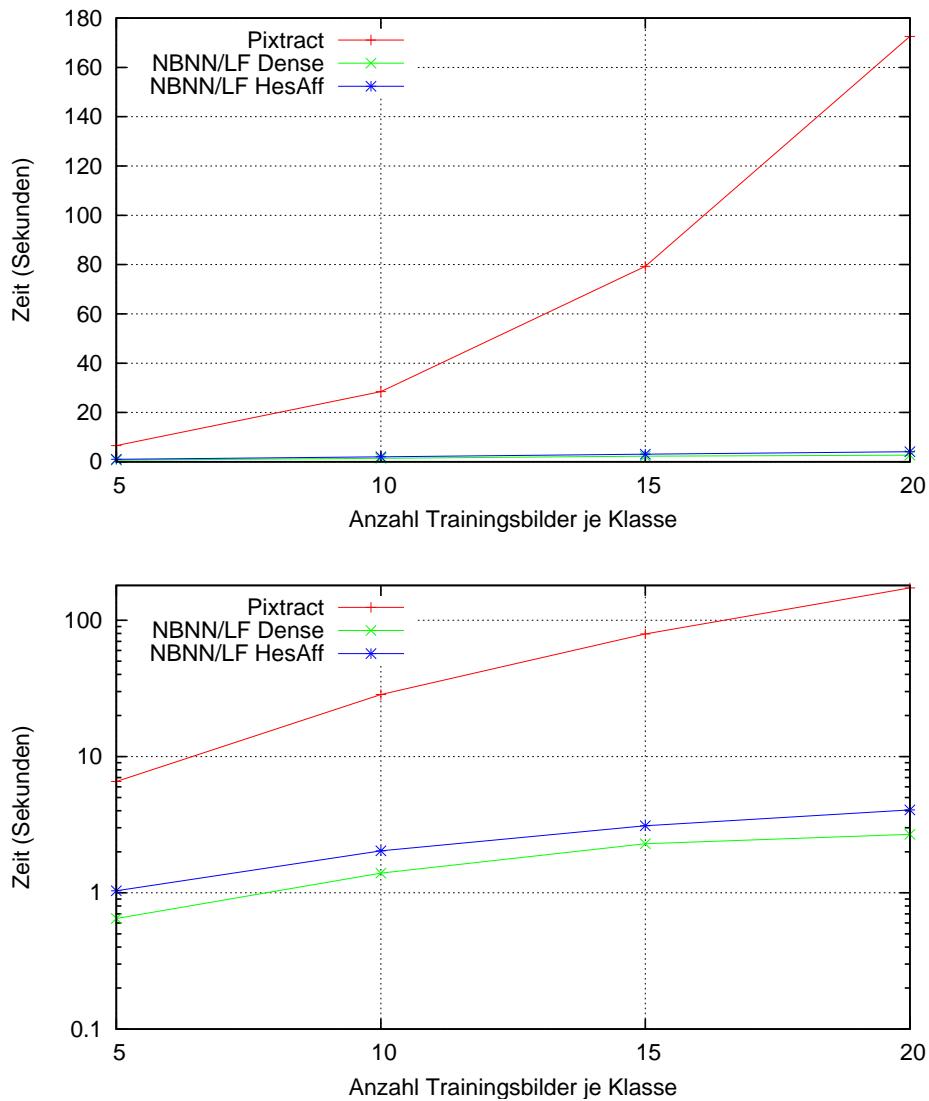


Bild 9.3: Vergleich der CPU-Zeit für die Aufnahme einer neuen Klasse bei unterschiedlichen Anzahlen von Trainingsbildern in Pixtract sowie mit den Verfahren aus [BSI08] und [AF10]. Berücksichtigt wurde die durchschnittliche Zeit zwischen der Entgegennahme von Bildern in ihrem Rohformat und der Aufnahme der neuen Klasse ins Objektverzeichnis. Bei Pixtract wurde keine Säuberung der Trainingsbilder gemäß Abschnitt 7.4.1 unternommen.

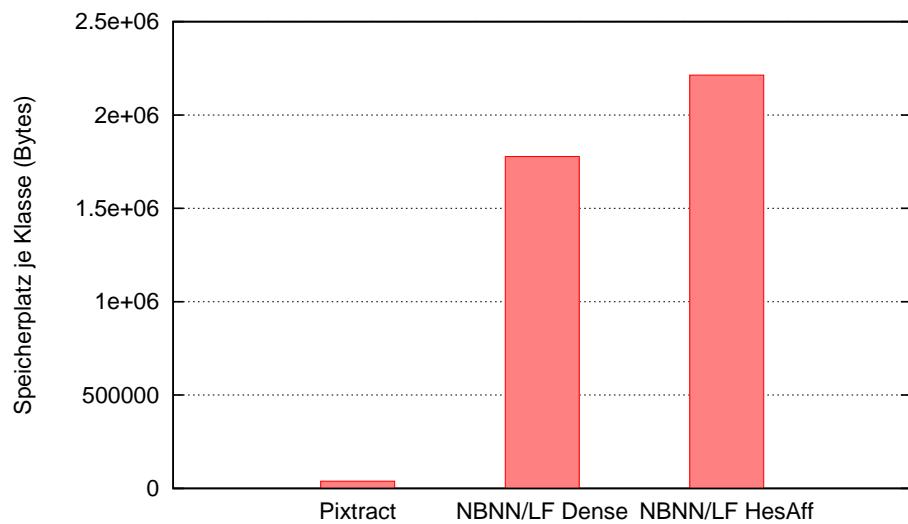


Bild 9.4: Vergleich des durchschnittlich benötigten Speicherplatzes für die Aufnahme einer neuen Klasse bei 20 Trainingsbildern in Pixtract sowie mit den Verfahren aus [BSI08] und [AF10].

9.5 Zusammenfassung

In diesem Abschnitt wurde das in der vorliegenden Arbeit entwickelte erweiterbare Verfahren Pixtract mit den erweiterbaren Ansätzen aus [BSI08] und [AF10] verglichen. Die Evaluation bezog sich auf die Genauigkeit der Klassifikation gemessen durch das Bewertungsmaß MAP, die Antwortzeit bei der Klassifikation eines Bildes, die Skalierbarkeit sowie die Erweiterbarkeit der Verfahren. Die Vorteile von Pixtract konnten in allen genannten Dimensionen belegt werden.

Dank der optimalen Kombination der Merkmale in Pixtract konnte in Abschnitt 9.2 sowohl eine bessere Genauigkeit bei der Klassifikation, als auch eine bessere Skalierbarkeit auf eine hohe Gesamtanzahl an Klassen für Pixtract festgestellt werden. Da Pixtract bei der Klassifikation nicht direkt auf den SIFT-Deskriptoren arbeitet, sondern kompakte Repräsentationen der einzelnen Klassen einsetzt (u. a. meanGIST und klassenspezifische Vokabulare, vgl. Kapitel 6), ist die Antwortzeit bei der Klassifikation eines Bildes in Abschnitt 9.3 bei Pixtract wesentlich kürzer, als bei den anderen in der Evaluation berücksichtigten Ansätzen. Die Berechnung der kompakten Beschreibungen der einzelnen Klassen erfolgt bei der Aufnahme von einer neuen Klasse. Dies führt bei der Erweiterung des Objektverzeichnisses zu längeren Berechnungszeiten im Vergleich mit den Verfahren aus [BSI08] und [AF10]. Die durchschnittliche Zeit von 3 Minuten für das Hinzufügen einer neuen Klasse (bei der Verwendung von 20 Trainingsbildern) in Pixtract ist – insbesondere im Vergleich zu den Zahlen bei SVM-basierten Ansätzen – jedoch durchaus vertretbar, da dieser Wert unabhängig von der Anzahl der Klassen im Objektverzeichnis ist.

Im Anschluss an den Vergleich von Pixtract mit anderen erweiterbaren Ansätzen, werden im folgenden Kapitel die Erkenntnisse dieser Arbeit zusammengefasst und weitere Möglichkeiten zur Verbesserung der Objekterkennung in Bildern beschrieben.

KAPITEL 10

ZUSAMMENFASSUNG UND AUSBLICK

Die Flut an digitalen Bildern nimmt Jahr für Jahr stetig zu. Nach einer aktuellen Schätzung von [SS11] werden jährlich 50 Milliarden digitale Fotoaufnahmen erstellt. Um spezifische Bilder auch nur in einem Bruchteil dieser enormen Menge wiederzufinden, müssen die Aufnahmen entsprechend indiziert werden. Bei der Bildsuche ist heute weiterhin die textbasierte Methode am weitesten verbreitet, grafische Verfahren konnten sich bislang nicht durchsetzen. Dementsprechend müssen die Bilder in Text übersetzt werden. Diese textuelle Beschreibung von Bildern unterstützt nicht nur die textbasierte Suche, sondern erleichtert auch der stetig wachsenden Gruppe von Menschen mit Sehbehinderungen das alltägliche Leben sowie den Umgang mit moderner Technik. Die Indizierung kann bei der immensen Anzahl an Bildern nicht mehr manuell durch Menschen erledigt werden und muss so gut wie möglich automatisch erfolgen. Im Gegensatz zur Indizierung von Bildern im Web, die meistens in einen relevanten textuellen Kontext eingebettet sind, kann bei der automatischen Annotation nur auf den Inhalt der Bilder sowie auf die von der Digitalkamera bei der Erzeugung der Fotoaufnahme annotierten technischen Metadaten zugegriffen werden.

Für die automatische Annotation der Bilder ist eine Wissensbasis nötig, der Muster der realen Welt bekannt sind, und die, sofern sie in Fotos erkannt wurden, diese in Textform übersetzen kann. Dieser Datenbestand der dem System bekannten Objekte wird niemals alle Objekte beinhalten und wird dadurch mit voranschreitender Zeit stetig

wachsen. Somit muss das Annotationssystem sowohl neue Objekte hinzulernen als auch bei einer großen Anzahl von bekannten Klassen weiterhin gute Annotationen erstellen können. Das primäre Ziel ist es ein Annotationssystem zu entwickeln, welches ähnlich wie Menschen ca. 30 000 Objekte¹ auf Anhieb unterscheiden und benennen kann. Leider vernachlässigen aktuelle Verfahren zur automatischen Annotation von Bildern sowohl den Aspekt der Erweiterbarkeit als auch die Skalierbarkeit der Systeme.

Vor diesem Hintergrund wurde in dieser Dissertation die automatische Annotation von Bildern als ein erweiterbares System unter dem Namen Picture Annotation Extraction (Pixtract) realisiert. Um sowohl die textbasierte Suche als auch die textuelle Beschreibung von Bildern für Menschen mit Sehbehinderungen optimal unterstützen zu können, wurden in Kapitel 3 die Anforderungen an die textuelle Annotation von Bildern ermittelt. Zusätzlich wurden unterschiedliche logische Strukturierungen für Bildannotationen verglichen und vorhandene Metadatenstandards für Bilder betrachtet. Basierend auf der Analyse wurde festgestellt, dass für die Beschreibung von Bildern die Erkennung von Objekten am wichtigsten ist.

In Kapitel 4 wurden verschiedene aktuelle Ansätze für die manuelle, semi- und vollautomatische Annotation von Bildern verglichen. Aufbauend auf die Anforderung Objekte in Bildern zu identifizieren, wurden insbesondere Verfahren zur Objekterkennung in Bildern vorgestellt. Unter diesen Ansätzen etablierte sich in den letzten Jahren die Bag-of-Words-Methode (BoW) als bestes Verfahren. BoW stammt ursprünglich aus dem Text-Mining-Umfeld, wo es für die automatische Ermittlung eines Wörterbuchs aus zwei identischen Texten in unterschiedlichen Sprachen konzipiert wurde. Im Bereich von visuellen Medien, wie z. B. Bildern oder Videos, werden die Wörter durch sog. visuelle Wörter (interessante Punkte und deren Umgebung beschrieben durch robuste Merkmale) ersetzt. Das Wörterbuch besteht dementsprechend aus der Abbildung von einer Menge von visuellen Wörtern auf textuelle Wörter. Im Anschluss kann das aus einer Stichprobe erlernte Wörterbuch auf neue Bilder angewendet werden, die somit in Text „übersetzt“ werden.

Es ist naheliegend, dass nur diejenigen Objekte erkannt werden können, welche dem Annotationssystem bekannt sind. Der Datenbestand wird sich auch niemals auf alle Objekte der Welt beziehen können und muss somit leicht erweiterbar sowie auf eine große Anzahl von Klassen skalierbar sein. Nahezu alle aktuellen Verfahren zur Objekterkennung sind

¹ Schätzung nach [Bie87]

durch die Wahl der verwendeten Merkmale oder der eingesetzten Klassifikationsmethode stark bzgl. der Erweiterbarkeit beschränkt. Kapitel 5 beschreibt die Erweiterbarkeitsprobleme auf verschiedenen Stufen des Klassifikationsvorgangs und identifiziert begehbarer Lösungswege. Dabei beeinflussen viele Parameter die Objekterkennung und dadurch die Genauigkeit der Annotation. Eine möglichst optimale Kombination diesbezüglich wurde in Kapitel 6 anhand von mehreren großen und etablierten Datensätzen ermittelt. Zusätzlich wurde auch die Möglichkeit der Einschränkung des Suchraums und dadurch die Verschnellerung des Annotationsvorgangs zu Verlusten bei der Genauigkeit der Objekterkennung in Relation gesetzt.

Das erweiterbare Verfahren wurde im Pixtract-Framework umgesetzt, welches in Kapitel 7 im Detail vorgestellt wurde. Wesentliche Eigenschaft des Frameworks ist die Trennung zwischen der merkmalsbasierten Suche zur Annotation von Bildern und der textbasierten Suche zum Auffinden von bereits annotierten Bildern. Kernstück von Pixtract ist ein konfigurierbares und durch neue Merkmalsextraktoren leicht erweiterbares Werkzeug *FeatExt*, welches sowohl für das Erlernen neuer Klassen als auch für die Annotation von Bildern eingesetzt wird. Der Zugriff auf Pixtract wird durch eine Webschnittstelle ermöglicht. Für die neu zu erlernenden Klassen, deren Bilder die Anwender zusammenstellen und an Pixtract übertragen, wurde als Unterstützung ein einfacher Filter umgesetzt, welcher unsaubere Bilder ausschließt. Zusätzlich wurde auch eine an MusicID angelehnte PicID-Schnittstelle zur Verfügung gestellt, welche in externen Bildverwaltungsprogrammen oder textbasierten Suchdiensten – wie z. B. Lucene – integriert werden kann und die Annotation von Bildern direkt im Programm ermöglicht.

Zur Anreicherung und Verbesserung der Bildannotation wurde in Kapitel 8 die Verwendbarkeit von durch OCR-Verfahren ermitteltem Text in natürlichen Fotoaufnahmen untersucht. Im Rahmen dieser Evaluation wurde der NEOCR-Datensatz entwickelt und aktuelle OCR-Werkzeuge anhand dieses Datensatzes bewertet. Mit Bedauern musste festgestellt werden, dass die Genauigkeit der Erkennung von Text in natürlichen Fotoaufnahmen zur Zeit noch nicht ausreichend ist, als dass sie für die Bildannotation sinnvoll verwendet werden könnte.

Aus der Vielzahl an Systemen für die Objekterkennung wurden einige wenige identifiziert, welche sich grundsätzlich für die Erweiterung durch neue Klassen eignen. In Kapitel 9 wurde Pixtract mit diesen Verfahren bzgl. der Erweiterbarkeit, Skalierbarkeit und Genauigkeit verglichen und in allen Evaluationen als besser befunden. Auch im Vergleich mit Messungen für mehrere 1 000 Klassen in [DBLFF10] schneidet Pixtract

bei einer Datenbasis von mehr als 3 000 Klassen im Bereich des Stands der Technik ab – und das trotz der einfachen Erweiterbarkeit des Systems.

Ausblick

Mit Pixtract wurde ein effizientes erweiterbares System zur Annotation von Bildern erstellt. Die Genauigkeit der Bildannotationen reicht zur Zeit jedoch leider noch nicht an die des Menschen heran. Für die weitere Verbesserung der Annotationen können mehrere Forschungsrichtungen verfolgt werden. Im nachfolgenden werden einige Wege und damit mögliche Folgearbeiten dieser Dissertation kurz erläutert.

Einsatz von semantischen Hierarchien

Der Großteil der bisherigen Forschung auf dem Gebiet der Objekterkennung verwendete lediglich unstrukturierte textuelle Vokabulare für die Zuordnung von Bildern zu Konzepten. Begünstigt wurde diese Entwicklung durch zahlreiche Datensätze mit flacher Annotation und einer einfachen Bewertung der Objekterkennung mittels der Maße AUC und MAP. Der Nachteil jedoch ist, dass unscharfe oder ungenaue Zuordnungen stark benachteiligt werden. Des Weiteren ist auch keinerlei Flexibilität bzgl. der semantischen Ebene der Annotation gegeben.

Untersuchungen in [GP08] und [DBLFF10] zeigten, dass zwischen der Struktur von semantischen Hierarchien (wie z. B. WordNet) und der visuellen Ähnlichkeit bzw. Verwechslungen einzelner Klassen signifikante Zusammenhänge existieren. Erste Schritte für die Integration des Wissens in semantischen Hierarchien in die Klassifikation wurden u. a. in [ZW07] und [MS07] unternommen. Trotz des hohen Potenzials zur Verbesserung der Bildannotationen durch den sinnvollen Einsatz von Ontologien, Taxonomien oder semantischen Netzwerken kommt die Forschung nur langsam voran. Der Grund liegt zum Einen in der eingeschränkten Verfügbarkeit von entsprechend annotierten Datensätzen. Einen Anfang stellt das Projekt ImageNet [DDS⁺09] dar, welches kontinuierlich eine Bilddatenbasis basierend auf den Konzepten von WordNet aufbaut. Weitere müssen jedoch folgen – insbesondere Datensätze, die Bilder mit mehr als nur einem Objekt enthalten, um das volle Potenzial von strukturierten Vokabularen auszunutzen zu können. Ein weiterer Grund ist auch das Fehlen eines einheitlich akzeptierten Bewertungsmaßes für

die Evaluation. Möglichkeiten wurden in Abschnitt 4.1.4.3 vorgestellt, ein Einverständnis über das am besten geeignete Bewertungsmaß existiert jedoch zur Zeit noch nicht.

Zusätzlich bleiben auch die Fragen offen, welche semantische Hierarchie eingesetzt werden soll und welche Beziehungstypen zwischen den Konzepten der Hierarchie (siehe Anhang B) sinnvoll für die Verbesserung der Klassifikation und Annotation sind. Im Bereich der Objekterkennung wird überwiegend WordNet unter Verwendung der Hyperonymbeziehungen eingesetzt. Die sinnvolle Integration von komplexen semantischen Netzwerken wie z. B. ConceptNet¹ [LS04] oder DBpedia² [BLK⁺09] steht bislang noch aus.

Annotationsqualität

Sowohl manuell als auch automatisch erstellte Bildannotationen können ungenau oder mit Störungen versehen sein. Die Ursache der Ungenauigkeit bzw. Störung kann einerseits in der Quelle der Annotation liegen. Manuelle Annotationen können durch einen ausgebildeten Annotator oder durch eine Menge von Benutzern (sog. „crowd“) erfolgen. Automatisch ermittelte Annotationen können aus dem umliegenden Text, aus Annotationen von ähnlichen Bildern und direkt aus dem Inhalt des Bildes ermittelt werden.

Andererseits ist die Ungenauigkeit einer Annotation auch von den Anforderungen der Anwendung bzw. der Vorstellung des Benutzers bzgl. seines Aufgabengebietes abhängig. Beispielsweise ist bei der Suche nach einem passenden Bild für einen Zeitungs- oder Webartikel der Detaillierungsgrad der Bildannotation maßgebend. Ein erster Schritt in diese Richtung wurde in dieser Dissertation durch die Analyse der Anforderungen an die Bildannotation aus der Sicht der textbasierten Suche und von Menschen mit Sehbehinderungen unternommen.

Auf dem Gebiet die Qualität von Bildannotationen oder Annotationen generell zu fassen gibt es bislang nur wenige Arbeiten. [FYS⁺09] unterscheidet Synonyme, mehrdeutige, Spam, unvollständige und zu weit gefasste Tags. In [JSW11] wurden durch die Befragung von 35 Personen (Studenten und Mitarbeiter, somit also keine ausgebildeten Annotatoren)

1 <http://conceptnet5.media.mit.edu/>

2 <http://dbpedia.org>

15 grobe Kriterien identifiziert¹, nach denen ein Term bei der Beschreibung von Bildern von den Benutzern als nützlich eingestuft wurde.

Es ist zu untersuchen in wie weit die Datenqualitätsdimensionen nach [SMB05] und [SGTS07] auf die Qualität von Bildannotationen übertragbar sind. Wie ist z.B. die Genauigkeit einer Bildannotation zu fassen? Grundlegend kann zwischen syntaktischer und semantischer Genauigkeit unterschieden werden. Syntaktische Fehler können durch den Einsatz eines kontrollierten Vokabulars behoben bzw. umgangen werden. Bei der semantischen Genauigkeit sind jedoch mehrere Fragen zu beantworten, unter anderem:

- Soll die Annotation für das gesamte Bild oder für einzelne Bildregionen erstellt werden?
- Wie genau sollen Bildregionen erfasst werden (genaue Umrandung, Polygon oder Rahmen)?
- Wie sind Überlappungen von Bildregionen bzgl. der Genauigkeit der Bildannotation zu interpretieren?

Für die weiteren Qualitätsdimensionen ergeben sich ähnliche Fragestellungen, sofern sie auf Bildannotationen angewendet werden sollen. Ggf. ist auch die Definition von neuen Qualitätsdimensionen spezifisch für Bildannotationen notwendig, wie sie auch bereits in [JSW11] angedeutet wurden. Aufbauend auf der einheitlichen Definition der Annotationsqualität müssen anschließend entsprechende Bewertungsmaße definiert werden, um die Qualität von Bildannotationen messen und vergleichen zu können.

Einbeziehung von sichtbarem Text

Natürliche Fotoaufnahmen enthalten in vielen Fällen Text. Zum Teil gibt dieser Text Aufschluss darüber, in welchem Kontext das Foto aufgenommen wurde bzw. welche Objekte auf dem Bild zu sehen sind. Beispielsweise kann bei einer Fotoaufnahme von einem Marktstand ein Gemüse durch den Text auf der vorhandenen Tafel eindeutig identifiziert werden, auch wenn die visuelle Erkennung der Objekte fehlschlägt. Um den in Bildern enthaltenen Text für die Verbesserung der Klassifikation und der Annotation nutzen

¹ Die 15 identifizierten Kriterien nach [JSW11] sind: Informativeness, Relevance, Connecting to the image, Accuracy, Specificity, Descriptiveness, Level of Detail, Personalization, Ease of Use, Importance, Generality, Redundancy, Objectiveness, Moods und Structure.

zu können, müssen jedoch mehrere Fragen beantwortet und zur Zeit noch ausstehende Probleme gelöst werden.

Den ersten Schritt hierzu stellt die ausreichend gute Erkennung und Extraktion von Text in Bildern dar. Die Auswertungen in Kapitel 8 haben verdeutlicht, dass keines der aktuell verfügbaren OCR-Werkzeuge Text in natürlichen Fotoaufnahmen ausreichend gut erkennt. Anschließend muss ein Verfahren entwickelt werden, was die Relevanz von im Bild vorhandenem Text im Bezug auf den sonstigen Bildinhalt bewertet. Hilfreiche Faktoren können dabei u. a. die Größe der Textregion relativ zur Bildgröße, die Größe der Schriftzeichen relativ zu anderen Texten im Bild sowie die mittlere Entfernung zu einem Objekt oder zu Objektgruppen sein. Zuletzt muss der erkannte Text in die Klassifikation sinnvoll eingebunden werden, so dass als Ergebnis die Annotation verbessert wird. Hierzu können sowohl semantische Hierarchien als auch Ansätze aus der Analyse vom Bildkontext herangezogen werden.

Kontextanalyse in Bildern

Eine Vielzahl von Ansätzen verfolgt das Ziel die semantische Lücke zwischen Merkmalen (wie Farben, Formen oder Texturen) und Konzepten (wie Objekte, Personen, Ereignisse oder Orte) zu schließen. Dabei streben viele Verfahren die Erkennung von nur einem Teilbereich von Konzepten an. Sowohl für die Verbesserung der Klassifikation als auch für eine genauere Annotation kann es hilfreich sein den Kontext innerhalb eines Bildes mit einzubeziehen. Beziehungen zwischen identifizierten Objekten, Personen und Textbereichen im Bild können Hinweise zur Relevanz und zur genauen Beschaffenheit der einzelnen Bildelemente geben. Des Weiteren könnten ausgehend von einer Wissensbasis, Erkenntnissen der Neuropsychologie sowie basierend auf erkannten Objekten, Personen, Text und sonstigen verfügbaren Metadaten (wie z.B. Zeit und Ort) im Bild dargestellte Ereignisse oder Aktivitäten, ggf. sogar Gefühle abgeleitet werden. Einen ersten Schritt in diese Richtung stellt das Buch [DLST09] dar, welches die wichtigsten Errungenschaften des letzten Jahrzehnts sowohl aus dem Gebiet der Informatik als auch aus dem Bereich der Neurowissenschaften zusammenstellt. Die Integration der verschiedenen Wissensquellen und die Untersuchung von Querbeziehungen ist zwar äußerst komplex, hält jedoch ein unausschöpfliches Potenzial für wesentliche Verbesserungen bzgl. des rechnergestützten Bildverständnisses bereit.

Anhang

ANHANG A

RELEVANTE FELDER AUS METADATENSTANDARDS

In diesem Anhang werden die für inhaltsbeschreibende textuelle Bildannotationen relevanten Felder aus den Metadatenstandards IPTC und XMP aufgelistet und deren primärer Verwendungszweck kurz erläutert. Weitere Informationen zu den erwähnten Metadatenstandards sind in Abschnitt 3.3.1 zu finden.

A. Relevante Felder aus Metadatenstandards

ATTRIBUTNAME	DATENTYP	BEDEUTUNG
AddlModelInfo	Text	Ergänzende Information zum Foto-modell
ArtworkOrObject	Menge von ArtworkOrObjectDetails	Kunstwerke oder Objekte im Bild, Beschreibung des Datentyps siehe Tabelle A.2
CVTerm	CV-Code	Menge von Termen aus einem kontrollierten Vokabular
Event	Text	Benennt oder beschreibt das Ereignis oder die Aktion im Bild
LocationShown	Menge von Location-Details	Beschreibung des Ortes, welches im Bild zu sehen ist, Beschreibung des Datentyps siehe Tabelle A.3
ModelAge	Menge von Ganzzahlen	Alter der menschlichen Modelle im Bild
OrganisationInImageCode	CV-Code	Code aus einem kontrollierten Vokabular für die Organisation die im Bildinhalt vertreten ist
OrganisationInImageName	Text	Name der Organisation die im Bildinhalt vertreten ist
PersonInImage	Text	Name der Personen im Bild

Tabelle A.1: Bildbeschreibende Attribute aus dem IPTC Extension Schema [IPT08].

ATTRIBUTNAME	DATENTYP	BEDEUTUNG
AOCopyrightNotice	Text	Enthält Informationen zum Urheberrecht
AOCreator	Text	Name der Person, der das abgebildete Kunstwerk oder das Objekt erstellt hat
AODateCreated	Datum	Datum, an dem das Kunstwerk oder das Objekt im Bild erstellt wurde
AOSource	Text	Organisation, die das Kunstwerk oder das Objekt besitzt und registriert hat
AOSourceInvNo	Text	Inventarnummer des Kunstwerks oder des Objekts
AOTitle	Text	Titel des Kunstwerks oder des Objekts

Tabelle A.2: Attribute des Datentyps ArtworkOrObjectDetails nach [IPT08].

ATTRIBUTNAME	DATENTYP	BEDEUTUNG
City	Text	Name der Stadt
CountryCode	ISO-Code	ISO-Code des Landes
CountryName	Text	Name des Landes
ProvinceState	Text	Name der Provinz oder des Bundesstaates innerhalb des Landes
Sublocation	Text	Name eines Stadtteils oder eines Monuments in der Nähe der Stadt
WorldRegion	Text	Name der Weltregion

Tabelle A.3: Attribute des Datentyps LocationDetails nach [IPT08].

A. Relevante Felder aus Metadatenstandards

ATTRIBUTNAME	DATENTYP	BEDEUTUNG
dc:description	Text	Beschreibung des Inhalts, mehrere Beschreibungen in verschiedenen Sprachen sind möglich
dc:relation	Text	Beziehungen zu anderen Dokumenten
dc:subject	Text	Inhaltsbeschreibende Stichwörter
dc:title	Text	Titel des Bildes, mehrere Titel in verschiedenen Sprachen sind möglich
xmp:identifier	Text	Feld von Strings, welches das Dokument innerhalb eines gegebenen Kontexts eindeutig identifiziert
xmp:label	Text	Wort oder kurzer Ausdruck der ein Dokument einer benutzerdefinierten Gruppe zuweist
xmp:nickname	Text	Kurzer informeller Name des Dokuments
photoshop:category	Text	Kategorie gekennzeichnet durch 3 7-Bit ASCII-Zeichen
photoshop:supplementalcategories	Text	Ergänzende Kategorien
prism:alternatetitle	Text	Alternativer Titel für das Dokument
prism:event	Text	Ereignis oder die Aktion verknüpft mit dem Dokumentinhalt
prism:industry	Text	Industrie oder Industriesektor, dem das Dokument zugeordnet ist
prism:keyword	Text	Inhaltsbeschreibende Stichwörter
prism:location	Text	Ort oder die Aktion verknüpft mit dem Dokumentinhalt
prism:object	Text	Objekte im Dokument, primär zur Klassifizierung von Produkten gedacht
prism:organisation	Text	Name der Organisation über welches sich das Dokument handelt
prism:person	Text	Name der Personen über die sich das Dokument handelt
prism:teaser	Text	Kurze Beschreibung des Dokuments

Tabelle A.4: Bildbeschreibende Attribute aus den XMP Schemata – Teil 1.

ATTRIBUTNAME	DATENTYP	BEDEUTUNG
disc:location	Text	Name der Stadt bzw. des Stadtteils, wo das Bild entstanden ist
disc:state	Text	Name der Provinz oder des Bundesstaates, wo das Bild entstanden ist
disc:country	Text	Name des Landes, wo das Bild entstanden ist
disc:subject	Text	Kurze Beschreibung des Bildinhalts
disc:caption	Text	Untertitel (Beschreibung) des Bildes
disc:keywords	Text	Inhaltsbasierte Stichwörter

Tabelle A.5: Bildbeschreibende Attribute aus den XMP Schemata – Teil 2.

ANHANG B

LINGUISTISCHE BEZIEHUNGEN ZWISCHEN WÖRTERN

In Tabelle B.1 werden die verschiedenen linguistischen Begriffe zur Beschreibung von semantischen Beziehungen zwischen Wörtern basierend auf [Mil95] erklärt. Die hier vorgestellten Beziehungen sind alle in die WordNet Ontologie integriert.

B. Linguistische Beziehungen zwischen Wörtern

BEZIEHUNG	KATEGORIE	BEDEUTUNG	ART	GEGENTEIL	BEISPIEL
Synonymie (Ähnlichkeit)	Substantiv, Verb, Adverb, Adjektiv	die Wörter haben eine ähnliche oder die gleiche Bedeutung	symmetrisch	Antonymie	Apfelsine – Orange, Dame – Frau
Antonymie (Gegensätzlichkeit)	Adjektiv, (Verb, Substantiv)	die Wörter haben eine entgegengesetzte Bedeutung	symmetrisch	Synonymie	trocken – nass, schnell – langsam
Hyperonymie (Überbegriff)	Substantiv	die Wörter stehen in einer hierarchischen Beziehung (Generalisierung); das Hyperonym ist der übergeordnete, allgemeinere Begriff	transitiv	Hyponymie	Säugetier – Katze – Tiger – bengalischer Tiger, Konstruktion – Gebäude – Wolkenkratzer
Hyponymie (Unterbegriff)	Substantiv	die Wörter stehen in einer hierarchischen Beziehung (Spezialisierung); das Hyponym ist der untergeordnete, speziellere Begriff	transitiv	Hyperonymie	siehe Beispiele für Hyponymie
Holonymie (Ganzes / Teil)	Substantiv	die Wörter stehen in einer hierarchischen Beziehung (Ganzes / Teil); das Holonym ist das Ganze	transitiv	Meronymie	Wald – Baum – Ast – Blatt Mensch – Arm – Hand – Finger
Meronymie (Teil / Ganzes)	Substantiv	die Wörter stehen in einer hierarchischen Beziehung (Teil / Ganzes); das Meronym ist das Teil	transitiv	Holonymie	siehe Beispiele für Holonymie
Troponymie (Art)	Verb	beschreibt die spezielle Art, wie etwas gemacht oder getan wird (Spezialisierung für Verbe)	transitiv	–	gehen – wandern, sprechen – vortragen
Implikation	Verb	Wenn-Dann-Beziehung zwischen Verben	transitiv	–	schnarchen – schlafen

Tabelle B.1: Semantische Beziehungen zwischen Wörtern nach [Mil95].

ANHANG C

ÄHNLICHKEITSMETRIKEN FÜR ONTOLOGIEBASIERTE BEWERTUNGSMASSE

In diesem Anhang werden ergänzend zu Abschnitt 4.1.4.3 weitere kantenbasierte, knotenbasierte und hybride Ähnlichkeitsmetriken für ontologiebasierte Ähnlichkeitsmaße kurz vorgestellt. Die Ähnlichkeitsmaße $sim(C_a, C_b)$ von Konzepten C_a und C_b können durch Umrechnung der Distanzmaße $dist(C_a, C_b)$ mittels $1 - dist(C_a, C_b)$ (nach [SZF07]) oder $\frac{1}{1+dist(C_a, C_b)}$ (nach [Lin98]) konvertiert werden. Anschließend können die Bewertungsmaße an der Stelle der *cost*-Funktion in Abschnitt 4.1.4.2 eingesetzt werden.

C.1 Kantenbasierte Ähnlichkeitsmaße

Die einfachste Möglichkeit kantenbasiert die Distanz zwischen zwei Konzepten zu bestimmen ist die Anzahl der Kanten $lenE(C_a, C_b)$ in der hierarchischen Ontologie vom Konzept C_a zum Konzept C_b aufzusummieren. In [Res95] wurde eine Variante eingeführt

(siehe Gleichung C.1), bei dem die Distanz in ein Ähnlichkeitsmaß umgewandelt wurde. D kennzeichnet dabei die Tiefe der hierarchischen Ontologie.

$$sim(C_a, C_b) = 2 \cdot D - lenE(C_a, C_b) \quad (C.1)$$

In [LC98] wurde für das in Gleichung C.2 vorgestellte Ähnlichkeitsmaß ebenfalls der kürzeste Pfad zwischen den Konzepten C_a und C_b , sowie die Tiefe D der hierarchischen Ontologie herangezogen.

$$sim(C_a, C_b) = -\log \left(\frac{lenE(C_a, C_b)}{2 \cdot D} \right) \quad (C.2)$$

Beide Ähnlichkeitsmaße haben den Nachteil, dass alle Kanten gleich gewichtet werden. Die in Abschnitt 4.1.4.2 aus [NL09] vorgestellte Kostenfunktion kann ebenfalls als kantenbasiertes Ähnlichkeitsmaß eingeordnet werden.

C.2 Knotenbasierte Ähnlichkeitsmaße

Einer der einfachsten Ansätze für ein knotenbasiertes Ähnlichkeitsmaß stammt von [WP94]. Dabei werden ähnlich zur Gleichung C.1 die Distanzen zwischen Knoten durch Aufsummieren ermittelt. In Gleichung C.3 wird statt der Anzahl der Kanten $lenE(C_a, C_b)$ die Anzahl der Knoten $lenN(C_a, C_b)$ berücksichtigt.

$$sim(C_a, C_b) = \frac{2 \cdot lenN(LCA(C_a, C_b), C_r)}{lenN(C_a, LCA(C_a, C_b)) + lenN(C_b, LCA(C_a, C_b)) + 2 \cdot lenN(LCA(C_a, C_b), C_r)} \quad (C.3)$$

Dabei markiert C_r den Wurzelknoten und $LCA(C_a, C_b)$ bezeichnet den nächsten gemeinsamen Elternknoten von den Konzepten C_a und C_b .

In [Res95] wurde neben der kantenbasierten Gleichung C.1 auch ein knotenbasiertes Ähnlichkeitsmaß definiert, welches den Informationsgehalt $IC(C_k) = -\log P(C_k)$ des Konzepts C_k berücksichtigt. $P(C_k)$ kennzeichnet dabei die Wahrscheinlichkeit für das Auftreten des Konzepts C_k . Die Ähnlichkeit zwischen den Konzepten C_a und C_b wird, wie in Gleichung C.4 dargestellt, durch den höchsten Informationsgehalt IC der nächsten gemeinsamen Elternknoten $LCA(C_a, C_b)$ bestimmt.

$$sim(C_a, C_b) = \max_{C_k \in LCA(C_a, C_b)} IC(C_k) \quad (C.4)$$

In [Lin98] wurde die Definition aus Gleichung C.4 erweitert. In Gleichung C.5 wird der Informationsgehalt der Ähnlichkeit zwischen zwei Konzepten mit dem Informationsgehalt der Konzepte selbst in Relation gesetzt.

$$sim(C_a, C_b) = \frac{2 \cdot IC(LCA(C_a, C_b))}{IC(C_a) + IC(C_b)} \quad (C.5)$$

[JC97] setzen in ihrem Ähnlichkeitsmaß die gleichen Größen in Beziehung, jedoch wird dabei die Differenz der Informationsgehalte berechnet:

$$sim(C_a, C_b) = 1 - (IC(C_a) + IC(C_b) - 2 \cdot IC(LCA(C_a, C_b))) \quad (C.6)$$

In [SDRL06] wurden die Ideen aus [Res95] und [Lin98] in einem Relevanz-Ähnlichkeit genannten Maß vereint. $P(LCA(C_a, C_b))$ kennzeichnet dabei die Wahrscheinlichkeit für das Auftreten des nächsten Elternknotens von Konzept C_a und C_b .

$$sim(C_a, C_b) = \left(\frac{2 \cdot IC(LCA(C_a, C_b))}{IC(C_a) + IC(C_b)} \cdot (1 - P(LCA(C_a, C_b))) \right) \quad (C.7)$$

Für den Vergleich von verschiedenen Objekterkennungsansätzen wurde beim ILSVRC 2011 Wettbewerb ein knotenbasiertes Ähnlichkeitsmaß eingesetzt, welches in Abschnitt 4.1.4.3 vorgestellt wurde.

C.3 Hybride Ähnlichkeitsmaße

In [SZF07] wurde ein Ähnlichkeitsmaß definiert, welches für jedes Konzept a priori Kosten berücksichtigt, die die erwarteten Kosten eines durchschnittlichen Benutzers erfassen. Die Berechnung dieser a priori Kosten basiert auf der Wahrscheinlichkeit ob ein Konzept einem anderem übergeordnet ist. Details zur Ableitung sind in [SZF07] zu finden. Als Endergebnis erhält man:

$$APS(C_k) = \left\lfloor \frac{1}{n+2} \right\rfloor \quad (C.8)$$

, wobei n der Anzahl aller Kinder von Konzept C_k entspricht. Da in einer hierarchischen Ontologie, bei dem ein Knoten mehrere Eltern haben kann, sich mehrere gemeinsame

Elternknoten $LCA(C_a, C_b)$ für Konzepte C_a und C_b ergeben können, wird im folgenden derjenige Knoten $C_k \in LCA(C_a, C_b)$ ausgewählt, welcher die Gleichung

$$\arg \max_{C_k} r(C_k) \cdot 2^{h_max(C_k)} \quad (\text{C.9})$$

erfüllt, wobei $r(C_k)$ der Anzahl der verschiedenen Pfade zur Ermittlung von $C_k \in LCA(C_a, C_b)$ und $h_max(C_k)$ der Anzahl der Kanten auf dem längsten Pfad vom Wurzelknoten C_r zum Konzept C_k entspricht.

Zur weiteren Berechnung werden die Beziehungen zwischen den beteiligten Knoten nach deren Richtung unterschieden. Demnach ergibt sich der Koeffizient $\alpha(C_a, C_k)$ für die Richtung, dass ein Knoten C_a einen (nicht zwangsweise direkten) Elternknoten C_k hat durch die Gleichung C.10a. Für die Gegenrichtung ergibt sich der Koeffizient $\beta(C_b, C_k)$ nach der Gleichung C.10b, wobei C_b ein (nicht zwangsweise direkter) Kindknoten von C_k ist.

$$\alpha(C_a, C_k) = \frac{APS(C_k)}{APS(C_a)} \quad (\text{C.10a})$$

$$\beta(C_b, C_k) = APS(C_b) - APS(C_k) \quad (\text{C.10b})$$

Anschließend werden die von Konzept C_a nach C_b transferierten Kosten mit der Distanz zwischen den Konzepten in Relation gesetzt, wodurch sich das Distanzmaß in Gleichung C.11 ergibt. $\max Dist$ entspricht dabei der größten Distanz zwischen den zwei entferntesten Konzepten der gesamten Ontologie. Das Distanzmaß kann, wie oben erwähnt, durch $1 - dist(C_a, C_b)$ in ein Ähnlichkeitsmaß $sim(C_a, C_b)$ umgewandelt werden.

$$dist(C_a, C_b) = \frac{\log(1 + 2 \cdot \beta(C_b, C_k)) - \log(\alpha(C_a, C_k))}{\max Dist} \quad (\text{C.11})$$

Nach der Evaluation in [SZF07] auf der WordNet-Ontologie und der GeneOntology schnitt im Vergleich mit menschlichen Bewertungen das Distanzmaß in Gleichung C.11 als Bestes ab.

ANHANG D

MATHEMATISCHE BERECHNUNGEN

D.1 Verzerrung

Die generelle Repräsentation einer perspektivischen Transformation ist laut [Wol94] folgendermaßen definiert:

$$[x',y',w'] = [u,v,w] \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (\text{D.1})$$

Durch Normalisierung des Skalierungswertes ($a_{33} = 1$) erhält man folgende Gleichungen für die Umwandlung der Ursprungskoordinaten (u_k, v_k) für $k = 0,1,2,3$:

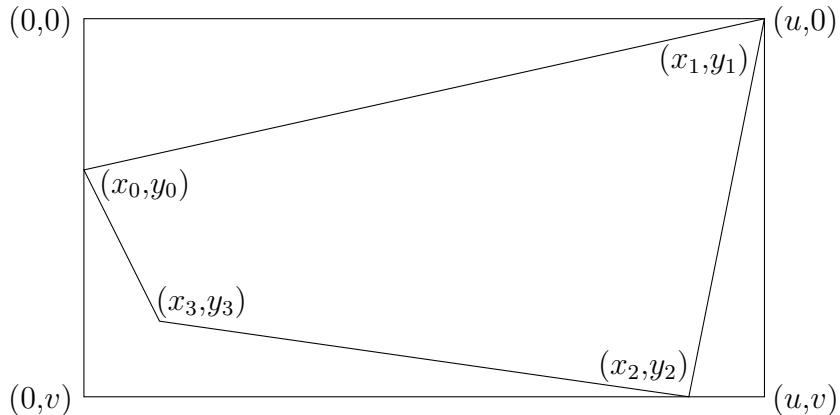
$$x = a_{11}u + a_{21}v + a_{31} - a_{13}ux - a_{23}vx \quad (\text{D.2a})$$

$$y = a_{12}u + a_{22}v + a_{32} - a_{13}uy - a_{23}vy \quad (\text{D.2b})$$

Es ergibt sich für die vier Koordinatenpaare folgendes lineares Gleichungssystem:

$$\begin{bmatrix} u_0 & v_0 & 1 & 0 & 0 & 0 & -u_0x_0 & -v_0x_0 \\ u_1 & v_1 & 1 & 0 & 0 & 0 & -u_1x_1 & -v_1x_1 \\ u_2 & v_2 & 1 & 0 & 0 & 0 & -u_2x_2 & -v_2x_2 \\ u_3 & v_3 & 1 & 0 & 0 & 0 & -u_3x_3 & -v_3x_3 \\ 0 & 0 & 0 & u_0 & v_0 & 1 & -u_0y_0 & -v_0y_0 \\ 0 & 0 & 0 & u_1 & v_1 & 1 & -u_1y_1 & -v_1y_1 \\ 0 & 0 & 0 & u_2 & v_2 & 1 & -u_2y_2 & -v_2y_2 \\ 0 & 0 & 0 & u_3 & v_3 & 1 & -u_3y_3 & -v_3y_3 \end{bmatrix} A = X \quad (\text{D.3})$$

Im vorliegenden Fall handelt es sich allerdings um eine Spezialform des Gleichungssystems, da eine Umwandlung eines Rechtecks in ein beliebiges, innenliegendes Viereck erfolgt, was in der folgenden Zeichnung ersichtlich ist und zu den darauffolgenden Gleichungen für die Transformation führt:



$$(0,0) \rightarrow (x_0, y_0)$$

$$(u,0) \rightarrow (x_1, y_1)$$

$$(u,v) \rightarrow (x_2, y_2)$$

$$(0,v) \rightarrow (x_3, y_3)$$

Durch Anwendung der vorherigen Regeln auf Gleichung D.3 erhält man folgende acht Gleichungen für die Transformationskomponenten:

$$a_{31} = x_0 \quad (\text{D.4a})$$

$$a_{32} = y_0 \quad (\text{D.4b})$$

$$a_{11}u + a_{31} - a_{13}ux_1 = x_1 \quad (\text{D.4c})$$

$$a_{12}u + a_{32} - a_{13}uy_1 = y_1 \quad (\text{D.4d})$$

$$a_{11}u + a_{21}v + a_{31} - a_{13}ux_2 - a_{23}vx_2 = x_2 \quad (\text{D.4e})$$

$$a_{12}u + a_{22}v + a_{32} - a_{13}uy_2 - a_{23}vy_2 = y_2 \quad (\text{D.4f})$$

$$a_{21}v + a_{31} - a_{23}vx_3 = x_3 \quad (\text{D.4g})$$

$$a_{22}v + a_{32} - a_{23}vy_3 = y_3 \quad (\text{D.4h})$$

Gleichung D.4a in D.4c:

$$a_{11}u + x_0 - a_{13}ux_1 = x_1 \quad \Rightarrow \quad a_{11} = (-x_0 + x_1 + a_{13}ux_1)/u \quad (\text{D.5})$$

Gleichung D.4b in D.4d:

$$a_{12}u + y_0 - a_{13}uy_1 = y_1 \quad \Rightarrow \quad a_{12} = (-y_0 + y_1 + a_{13}uy_1)/u \quad (\text{D.6})$$

Gleichung D.4a in D.4g:

$$a_{21}v + x_0 - a_{23}vx_3 = x_3 \quad \Rightarrow \quad a_{21} = (-x_0 + x_3 + a_{23}vx_3)/v \quad (\text{D.7})$$

Gleichung D.4b in D.4h:

$$a_{22}v + y_0 - a_{23}vy_3 = y_3 \quad \Rightarrow \quad a_{22} = (-y_0 + y_3 + a_{23}vy_3)/v \quad (\text{D.8})$$

Gleichungen D.4a, D.5, D.7 in Gleichung D.4e:

$$\begin{aligned} & (-x_0 + x_1 + a_{13}ux_1) + (-x_0 + x_3 + a_{23}vx_3) + x_0 - a_{13}ux_2 - a_{23}vx_2 = x_2 \\ & -x_0 + x_1 - x_2 + x_3 + a_{23}v(x_3 - x_2) = a_{13}u(x_2 - x_1) \\ \Rightarrow \quad a_{13} &= \frac{-x_0 + x_1 - x_2 + x_3 + a_{23}v(x_3 - x_2)}{u(x_2 - x_1)} \quad (\text{D.9}) \end{aligned}$$

Gleichungen D.4b, D.6, D.8, D.9 in Gleichung D.4f:

$$\begin{aligned}
 & (-y_0 + y_1 + a_{13}uy_1) + (-y_0 + y_3 + a_{23}vy_3) + y_0 - a_{13}uy_2 - a_{23}vy_2 = y_2 \\
 & -y_0 + y_1 - y_2 + y_3 + a_{13}u(y_1 - y_2) = a_{23}v(y_2 - y_3) \\
 & (-y_0 + y_1 - y_2 + y_3)(x_2 - x_1) + (-x_0 + x_1 - x_2 + x_3)(y_1 - y_2) = \\
 & a_{23}v((y_2 - y_3)(x_2 - x_1) - (x_3 - x_2)(y_1 - y_2)) \\
 a_{23} &= \frac{(-y_0 + y_1 - y_2 + y_3)(x_2 - x_1) + (-x_0 + x_1 - x_2 + x_3)(y_1 - y_2)}{v((x_2 - x_1)(y_2 - y_3) - (x_3 - x_2)(y_1 - y_2))} \\
 \end{aligned} \tag{D.10}$$

Gleichungen D.4b, D.6, D.8 in Gleichung D.4f:

$$\begin{aligned}
 & (-y_0 + y_1 + a_{13}uy_1) + (-y_0 + y_3 + a_{23}vy_3) + y_0 - a_{13}uy_2 - a_{23}vy_2 = y_2 \\
 & -y_0 + y_1 - y_2 + y_3 + a_{13}u(y_1 - y_2) = a_{23}v(y_2 - y_3) \\
 \Rightarrow a_{23} &= \frac{-y_0 + y_1 - y_2 + y_3 + a_{13}u(y_1 - y_2)}{v(y_2 - y_3)} \tag{D.11}
 \end{aligned}$$

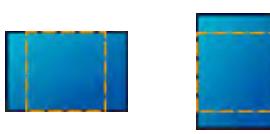
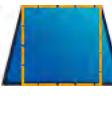
Gleichungen D.4a, D.5, D.7, D.11 in Gleichung D.4e:

$$\begin{aligned}
 & (-x_0 + x_1 + a_{13}ux_1) + (-x_0 + x_3 + a_{23}vx_3) + x_0 - a_{13}ux_2 - a_{23}vx_2 = x_2 \\
 & -x_0 + x_1 - x_2 + x_3 + a_{23}v(x_3 - x_2) = a_{13}u(x_2 - x_1) \\
 & (-x_0 + x_1 - x_2 + x_3)(y_2 - y_3) + (-y_0 + y_1 - y_2 + y_3)(x_3 - x_2) = \\
 & a_{13}u((x_2 - x_1)(y_2 - y_3) - (y_1 - y_2)(x_3 - x_2)) \\
 a_{13} &= \frac{(-x_0 + x_1 - x_2 + x_3)(y_2 - y_3) + (-y_0 + y_1 - y_2 + y_3)(x_3 - x_2)}{u((x_2 - x_1)(y_2 - y_3) - (x_3 - x_2)(y_1 - y_2))} \tag{D.12}
 \end{aligned}$$

Es ergibt sich nach Berechnung der Werte für obiges Beispiel also folgende Matrix

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \approx \begin{bmatrix} 0.36 & -0.22 & -0.07 \\ 0.24 & 0.56 & 0.04 \\ 0 & 2 & 1 \end{bmatrix}$$

die sich zusammensetzen lässt aus folgenden Einzelkomponenten [McC, Wol94]:

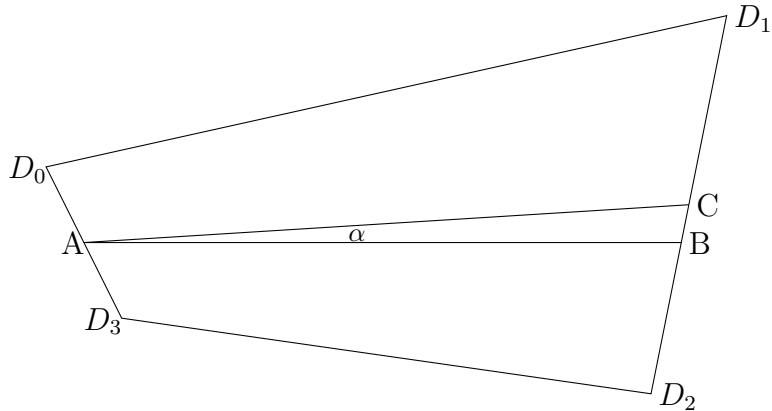
- Translationsmatrix: $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ T_u & T_v & 1 \end{bmatrix}$ 
- Matrix für Rotation: $\begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 
- Skalierungsmatrix: $\begin{bmatrix} S_u & 0 & 0 \\ 0 & S_v & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 
- Scherung entlang der x-Achse: $\begin{bmatrix} 1 & 0 & 0 \\ H_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 
- Scherung entlang der y-Achse: $\begin{bmatrix} 1 & H_u & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 
- perspektivische Verzerrung in x-Richtung: $\begin{bmatrix} 1 & 0 & P_u \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 
- perspektivische Verzerrung in y-Richtung: $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & P_v \\ 0 & 0 & 1 \end{bmatrix}$ 

Die Werte für die Translation (a_{31} und a_{32}) sind aus dem Bild sofort ersichtlich, da sie gleichwertig mit dem ersten Punkt des verzerrten Polygons sind. Die Art der Rotation lässt sich statt durch Berechnung von θ alternativ aus den beiden Werten für die Scherung (a_{12} und a_{21}) ablesen, da sie die Steigung der jeweiligen Gerade angeben. So kommt man durch den Arkustangens von a_{12} auf einen Winkel von $\approx -12^\circ$, da die Auslenkung nach oben geht, und für a_{21} erhält man eine Auslenkung zur y-Achse von $\approx 13^\circ$. Die Skalierungsfaktoren a_{11} und a_{22} sind nötig, da das Bild sonst durch die Rotation bzw. die beiden Scherungen seine Größe verändern würde. Letztendlich bleiben noch die beiden Werte für die perspektivische Verzerrung. a_{13} gibt durch den negativen Faktor eine Vergrößerung des Bildes an, je weiter man sich vom Ursprung entlang der x-Achse weg bewegt. Entsprechend wird das Polygon durch den positiven Wert von a_{23} in der Breite schmäler, je weiter man sich entlang der y-Achse nach unten begibt. Falls die gerade

genannten, für den „Distanzeffekt“ zuständigen, Faktoren a_{13} und a_{23} gleich null sind, handelt es sich statt einer perspektivischen um eine affine Transformation [Thy].

D.2 Rotation

Gegeben sind die vier Punkte des Verzerrungspolygons $D_k(x_k, y_k)$ für $k = 0, 1, 2, 3$ sowie die drei Punkte $A(x_4, y_4)$, $B(x_5, y_5)$ und $C(x_6, y_6)$ und gesucht ist $\alpha = \angle BAC$:



$$\overline{D_0D_3} = \sqrt{(|x_0 - x_3|)^2 + (|y_0 - y_3|)^2} \approx 2,2$$

$$\overline{D_1D_2} = \sqrt{(|x_1 - x_2|)^2 + (|y_1 - y_2|)^2} \approx 5,1$$

$$\overline{D_0D_1} = \sqrt{(|x_0 - x_1|)^2 + (|y_0 - y_1|)^2} \approx 9,2$$

$$\overline{D_2D_3} = \sqrt{(|x_2 - x_3|)^2 + (|y_2 - y_3|)^2} \approx 7,1$$

Falls $\overline{D_0D_3} = \overline{D_1D_2}$ und $y_0 = y_1$: $\alpha = 0$ (Rechteck)

Falls $\overline{D_0D_3} = \overline{D_1D_2}$ und $\overline{D_0D_1} = \overline{D_2D_3}$ (Parallelogramm): $\tan \alpha = \frac{|y_0 - y_1|}{|x_0 - x_1|}$

alle anderen Fälle (Trapez, beliebiges Viereck):

$$x_4 = (|x_3 - x_0|)/2 + \min(x_0, x_3) \quad y_4 = (|y_3 - y_0|)/2 + \min(y_0, y_3)$$

$$x_6 = (|x_2 - x_1|)/2 + \min(x_1, x_2) \quad y_6 = (|y_2 - y_1|)/2 + \min(y_1, y_2)$$

$$y_5 = y_4 \quad x_5 = ?$$

Schnittpunkt Gerade D_1D_2 : $y = y_1 + \frac{y_2-y_1}{x_2-x_1}(x - x_1)$ und Gerade AB :

$$\Rightarrow x_5 = \frac{(y_4 - y_1)(x_2 - x_1)}{y_2 - y_1} + x_1$$

Anwendung des Kosinussatzes:

$$\overline{AB} = x_5 - x_4$$

$$\overline{AC} = \sqrt{(|x_6 - x_4|)^2 + (|y_6 - y_4|)^2}$$

$$\overline{BC} = \sqrt{(|x_6 - x_5|)^2 + (|y_6 - y_5|)^2}$$

$$\cos \alpha = \frac{\overline{AB}^2 + \overline{AC}^2 - \overline{BC}^2}{2\overline{AB}\overline{AC}} \quad \text{falls } y_6 > y_5: \quad \alpha = 360 - \alpha$$

$\Rightarrow \alpha \approx 5.5^\circ$ für obiges Beispiel

ANHANG E

AUFLISTUNG VON DATEIINHALTEN

In diesem Anhang werden Beispiele für Inhalte verschiedener Dateien in Pixtract gezeigt.

Listing E.1 veranschaulicht den Aufbau einer XML-Parameterdatei für die Konfiguration von *FeatExt*. Eine von *FeatExt* erzeugte kurze XMP-Ausgabedatei ist in Listing E.2 zu sehen. Weitere Informationen zu *FeatExt* sind in Abschnitt 7.2.1 sowie in [Gal09] zu finden.

Listing E.3 zeigt die XML-Schema-Definition für die Metadaten des NEOCR Datensatzes. Die annotierten Metadaten für Abbildung 8.2 sind in Listing E.4 dargestellt. Weitere Informationen zum NEOCR Datensatz sind in Abschnitt 8.1 sowie in [Dic11, NDMW11b, NDMW11a] und [NDMW12] zu finden.

E.1 XML-Parameterdatei für FeatExt

```
1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2 <featext>
3   <settings>
4     <!-- Multithreading-Modus 1 = Eingeschaltet -->
5     <multithread>1</multithread>
6     <!-- Timeout in Sekunden -->
7     <plugintimeout>120</plugintimeout>
8     <!-- Debug-Modus 0 = Ausgeschaltet -->
9     <debug>0</debug>
10    <!-- Filter: Bilder welche die durch ; getrennten Zeichenketten
11       in dem Dateinamen enthalten werden ignoriert -->
12    <imgfilter>_canny;_sobel;_laplace;_cannyivt;_harris</imgfilter>
13    <!-- Skalieren auf (in Pixel) -->
14    <longside>800</longside>
15    <!-- Farbraum in dem gearbeitet wird -->
16    <colormodel>RGB</colormodel>
17    <!-- Layout der Logger-Ausgabe -->
18    <loglayout>%-5r %-8c %-5p - %m%n</loglayout>
19  </settings>
20  <!-- Bibliotheken-Plugins -->
21  <libraries>
22    <moments>
23      <name>Moments</name>
24      <class>Moments</class>
25      <namespace>moments</namespace>
26      <image></image>
27      <thread>1</thread>
28      <parameters>
29        <parameter>
30          <!-- Alle Werte die von 0 verschieden sind werden 1 -->
31          <name>Binary</name>
32          <index>0</index>
33          <value>0</value>
34          <defaultvalue>0</defaultvalue>
35        </parameter>
36        </parameters>
37      </moments>
38    </libraries>
39  <!-- Binaerdateien-Plugins -->
40  <binaries>
41    <mser>
42      <name>Maximally Stable Extremal Regions</name>
43      <class>MSER</class>
44      <namespace>mser</namespace>
45      <image></image>
46      <thread>1</thread>
47      <parameters>
48        <parameter>
49          <!-- Eingabebild -->
50          <name>Image</name>
```

```

51      <index>0</index>
52      <value>workdir/img.ppm</value>
53      <defaultvalue>workdir/img.ppm</defaultvalue>
54    </parameter>
55    <parameter>
56      <!-- Ausgabedatei -->
57      <name>Output</name>
58      <index>1</index>
59      <value>workdir/img.mser</value>
60      <defaultvalue>workdir/img.mser</defaultvalue>
61    </parameter>
62    <parameter>
63      <!-- Minimum size of output region -->
64      <name>Minimum_Size</name>
65      <index>2</index>
66      <value>30</value>
67      <defaultvalue>30</defaultvalue>
68    </parameter>
69    <parameter>
70      <!-- STDOUT Ausgabedatei -->
71      <name>Stdout</name>
72      <index>3</index>
73      <value>workdir/mser.log</value>
74      <defaultvalue>workdir/mser.log</defaultvalue>
75    </parameter>
76  </parameters>
77 </mser>
78 </binaries>
79 </featext>
```

Listing E.1: XML-Parameterdatei für die Konfiguration von *FeatExt*.

E.2 XMP-Ausgabedatei erstellt durch FeatExt

```

1 <x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="XMP Core 4.4.0">
2   <!-- Namensraum RDF -->
3   <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
4
5     <rdf:Description rdf:about="" 
6       <!-- Namensräume der MSER und Harris-Laplace-SIFT Plugins -->
7       xmlns:femser=
8       "http://ns.www6.informatik.uni-erlangen.de/fe/1.0/mser/"
9       xmlns:feharrlapsift=
10      "http://ns.www6.informatik.uni-erlangen.de/fe/1.0/harrlapsift/"
11    >
12    <femser>Data>
13      <!-- Daten des MSER Plugins -->
14      <rdf:Description
15        femser:Name="Maximally Stable Extremal Regions">
```

E. Auflistung von Dateiinhalten

```
16      <!-- Parameter des Feature-Detektors -->
17      <femser:Parameter
18          femser:Output_File="2"
19          femser:Ellipse_Scale="2"
20          femser:Image="temp/img.ppm"
21          femser:Output="temp/img.mser"
22          femser:Maximim_Relative_Area="0.010"
23          femser:Minimum_Size="30"
24          femser:Minimum_Margin="10"
25          femser:Stdout="temp/mser.log" />
26      <femser:FeatureData rdf:parseType="Resource">
27          <femser:Outputrdf:Seqrdf:li>392.155 147.213 ... 0.00685953</rdf:li>
31                  <rdf:li>390.145 149.645 ... 0.00298469</rdf:li>
32                  ...
33                  <rdf:li> 82.197 493.909 ... 0.01398453</rdf:li>
34          </rdf:Seq>
35          </femser:Output>
36      </femser:FeatureData>
37      </rdf:Description>
38  </femser:Data>
39  <feharrlapsift:Data>
40      <!-- Daten des Harris-Laplace-SIFT Plugins -->
41      <rdf:Description
42          <feharrlapsift:Name"="Harris Laplace">
43          <!-- Parameter des Feature-Detektors -->
44          <feharrlapsift:Parameter
45              <feharrlapsift:Harris_Laplace_Flag"="-harlap">
46              <feharrlapsift:Image"="temp/img.ppm">
47              <feharrlapsift:Hessian_Threshold"="200">
48              <feharrlapsift:Harris_Threshold"="10">
49              <feharrlapsift:Descriptor"="sift">
50              <feharrlapsift:Stdout"="temp/harrlapsift.log" />
51      <feharrlapsift:FeatureData rdf:parseType="Resource">
52          <feharrlapsift:Outputrdf:Seqrdf:li> 0 4 0 0 0 0 0 0 41 31 ... 0</rdf:li>
56                  <rdf:li>55 11 0 0 0 0 0 165 66 ... 21</rdf:li>
57                  ...
58                  <rdf:li>15 8 0 3 4 14 7 2 123 86 ... 89</rdf:li>
59          </rdf:Seq>
60      </feharrlapsift:Output>
61  </feharrlapsift:FeatureData>
62  </rdf:Description>
63  </feharrlapsift:Data>
64  </rdf:Description>
65  </rdf:RDF>
66 </x:xmpmeta>
```

Listing E.2: XMP-Ausgabedatei erstellt durch *FeatExt*.

E.3 NEOCR XML-Schema-Definition

```

1 i»<?xml version="1.0" encoding="UTF-8"?>
2 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
3   <xsd:element name="annotation">
4     <xsd:element name="filename" type="xsd:string"/>
5     <xsd:element name="folder" type="xsd:string"/>
6     <xsd:element name="source">
7       <xsd:complexType>
8         <xsd:sequence>
9           <xsd:element name="database" type="xsd:string"/>
10          <xsd:element name="software" type="xsd:string"/>
11          <xsd:element name="submittedBy" type="xsd:string" minOccurs="0"/>
12          <xsd:element name="author" type="xsd:string" minOccurs="0"/>
13          <xsd:element name="description" type="xsd:string" minOccurs="0"/>
14          <xsd:element name="originalFilename" type="xsd:string" minOccurs="0"/>
15        </xsd:sequence>
16      </xsd:complexType>
17    </xsd:element>
18    <xsd:element name="owner">
19      <xsd:complexType>
20        <xsd:sequence>
21          <xsd:element name="name" type="xsd:string"/>
22          <xsd:element name="chair" type="xsd:string"/>
23        </xsd:sequence>
24      </xsd:complexType>
25    </xsd:element>
26    <xsd:element name="properties">
27      <xsd:complexType>
28        <xsd:sequence>
29          <xsd:element name="width" type="xsd:positiveInteger"/>
30          <xsd:element name="height" type="xsd:positiveInteger"/>
31          <xsd:element name="depth" type="xsd:positiveInteger"/>
32          <xsd:element name="brightness">
33            <xsd:simpleType>
34              <xsd:restriction base="xsd:float">
35                <xsd:minInclusive value="0"/>
36                <xsd:maxInclusive value="255"/>
37              </xsd:restriction>
38            </xsd:simpleType>
39          </xsd:element>
40          <xsd:element name="contrast">
41            <xsd:simpleType>
42              <xsd:restriction base="xsd:float">
43                <xsd:minInclusive value="0"/>
44                <xsd:maxInclusive value="255"/>
45              </xsd:restriction>
46            </xsd:simpleType>
47          </xsd:element>
48        </xsd:sequence>
49      </xsd:complexType>
50    </xsd:element>
```

E. Auflistung von Dateiinhalten

```
51  <xsd:element name="object" minOccurs="1" maxOccurs="unbounded">
52    <xsd:complexType>
53      <xsd:sequence>
54        <xsd:element name="text" type="xsd:string"/>
55        <xsd:element name="deleted" type="xsd:boolean" default="0"/>
56        <xsd:element name="verified" type="xsd:boolean" default="0"/>
57        <xsd:element name="date" type="xsd:dateTime"/>
58        <xsd:element name="id" type="xsd:ID"/>
59        <xsd:element name="optical">
60          <xsd:complexType>
61            <xsd:sequence>
62              <xsd:element name="texture">
63                <xsd:simpleType>
64                  <xsd:restriction base="xsd:string">
65                    <xsd:enumeration value="low"/>
66                    <xsd:enumeration value="mid"/>
67                    <xsd:enumeration value="high"/>
68                  </xsd:restriction>
69                </xsd:simpleType>
70              </xsd:element>
71              <xsd:element name="brightness">
72                <xsd:simpleType>
73                  <xsd:restriction base="xsd:float">
74                    <xsd:minInclusive value="0"/>
75                    <xsd:maxInclusive value="255"/>
76                  </xsd:restriction>
77                </xsd:simpleType>
78              </xsd:element>
79              <xsd:element name="contrast">
80                <xsd:simpleType>
81                  <xsd:restriction base="xsd:float">
82                    <xsd:minInclusive value="0"/>
83                    <xsd:maxInclusive value="255"/>
84                  </xsd:restriction>
85                  <xsd:attribute name="inverted" use="optional" default="0" type="xsd
86                      :boolean"/>
87                </xsd:simpleType>
88              </xsd:element>
89              <xsd:element name="resolution" type="xsd:float"/>
90              <xsd:element name="noise">
91                <xsd:simpleType>
92                  <xsd:restriction base="xsd:string">
93                    <xsd:enumeration value="low"/>
94                    <xsd:enumeration value="mid"/>
95                    <xsd:enumeration value="high"/>
96                  </xsd:restriction>
97                </xsd:simpleType>
98              </xsd:element>
99              <xsd:element name="blurredness" type="xsd:float"/>
100             </xsd:sequence>
101           </xsd:complexType>
102         </xsd:element>
103         <xsd:element name="geometrical">
```

```

103      <xsd:complexType>
104          <xsd:sequence>
105              <xsd:element name="distortion">
106                  <xsd:complexType>
107                      <xsd:sequence>
108                          <xsd:element name="sx" type="xsd:float"/>
109                          <xsd:element name="sy" type="xsd:float"/>
110                          <xsd:element name="rx" type="xsd:float"/>
111                          <xsd:element name="ry" type="xsd:float"/>
112                          <xsd:element name="tx" type="xsd:float"/>
113                          <xsd:element name="ty" type="xsd:float"/>
114                          <xsd:element name="px" type="xsd:float"/>
115                          <xsd:element name="py" type="xsd:float"/>
116                  </xsd:sequence>
117          </xsd:complexType>
118      </xsd:element>
119      <xsd:element name="rotation">
120          <xsd:simpleType>
121              <xsd:restriction base="xsd:float">
122                  <xsd:minInclusive value="0"/>
123                  <xsd:maxExclusive value="360"/>
124              </xsd:restriction>
125          </xsd:simpleType>
126      </xsd:element>
127      <xsd:element name="arrangement">
128          <xsd:simpleType>
129              <xsd:restriction base="xsd:string">
130                  <xsd:enumeration value="horizontal"/>
131                  <xsd:enumeration value="vertical"/>
132                  <xsd:enumeration value="circular"/>
133              </xsd:restriction>
134          </xsd:simpleType>
135      </xsd:element>
136      <xsd:element name="cover">
137          <xsd:restriction base="xsd:unsignedByte">
138              <xsd:minInclusive value="0"/>
139              <xsd:maxInclusive value="100"/>
140          </xsd:restriction>
141          <xsd:attribute name="orientation" use="required">
142              <xsd:simpleType>
143                  <xsd:restriction base="xsd:string">
144                      <xsd:enumeration value="horizontal"/>
145                      <xsd:enumeration value="vertical"/>
146                  </xsd:restriction>
147              </xsd:simpleType>
148          </xsd:attribute>
149      </xsd:element>
150      </xsd:sequence>
151  </xsd:complexType>
152      </xsd:element>
153      <xsd:element name="typographical">
154          <xsd:complexType>
155              <xsd:sequence>

```

E. Auflistung von Dateiinhalten

```
156         <xsd:element name="font">
157             <xsd:simpleType>
158                 <xsd:restriction base="xsd:string">
159                     <xsd:enumeration value="handwriting"/>
160                     <xsd:enumeration value="standard"/>
161                     <xsd:enumeration value="special"/>
162                 </xsd:restriction>
163             </xsd:simpleType>
164         </xsd:element>
165         <xsd:element name="language"/>
166             <xsd:simpleType>
167                 <xsd:restriction base="xsd:string">
168                     <xsd:enumeration value="german"/>
169                     <xsd:enumeration value="english"/>
170                     <xsd:enumeration value="spanish"/>
171                     <xsd:enumeration value="hungarian"/>
172                     <xsd:enumeration value="italian"/>
173                     <xsd:enumeration value="latin"/>
174                     <xsd:enumeration value="french"/>
175                     <xsd:enumeration value="belgian"/>
176                     <xsd:enumeration value="russian"/>
177                     <xsd:enumeration value="turkish"/>
178                     <xsd:enumeration value="greek"/>
179                     <xsd:enumeration value="swedish"/>
180                     <xsd:enumeration value="czech"/>
181                     <xsd:enumeration value="portoguese"/>
182                     <xsd:enumeration value="numbers"/>
183                     <xsd:enumeration value="roman date"/>
184                     <xsd:enumeration value="abbreviation"/>
185                     <xsd:enumeration value="company"/>
186                     <xsd:enumeration value="person"/>
187                     <xsd:enumeration value="unknown"/>
188                 </xsd:restriction>
189             </xsd:simpleType>
190         </xsd:sequence>
191     </xsd:complexType>
192 </xsd:element>
193 <xsd:element name="difficult" type="xsd:boolean"/>
194 <xsd:element name="pt" minOccurs="4" maxOccurs="4">
195     <xsd:complexType>
196         <xsd:sequence>
197             <xsd:element name="x" type="xsd:positiveInteger"/>
198             <xsd:element name="y" type="xsd:positiveInteger"/>
199         </xsd:sequence>
200     </xsd:complexType>
201     </xsd:element>
202 </xsd:sequence>
203 </xsd:complexType>
204 </xsd:element>
205 </xsd:element>
206 </xsd:schema>
```

Listing E.3: XML-Schema für den NEOCR Datensatz.

E.4 XML-Annotation für ein Beispielbild aus dem NEOCR Datensatz

```

1 <annotation>
2   <filename>img_765077454.jpg</filename>
3   <folder>users/pixtract/dataset</folder>
4   <source>
5     <database>submitted</database>
6     <software>LabelMe Webtool</software>
7     <submittedBy>pixtract</submittedBy>
8     <author>Robert Nagy</author>
9     <description/>
10    <originalFilename>IMG_2600.JPG</originalFilename>
11  </source>
12  <owner>
13    <name>Anders Dicker</name>
14    <chair>FAU - CS 6 - Data Management</chair>
15  </owner>
16  <properties>
17    <width>3072</width>
18    <height>2304</height>
19    <depth>8/16</depth>
20    <brightness>105.808</brightness>
21    <contrast>45.5978</contrast>
22  </properties>
23  <object>
24    <text>FINSTERE GASSE</text>
25    <deleted>0</deleted>
26    <verified>0</verified>
27    <date>08-Oct-2010 10:19:56</date>
28    <id>0</id>
29    <optical>
30      <texture>mid</texture>
31      <brightness>164.493</brightness>
32      <contrast inverted="false">36.6992</contrast>
33      <resolution>49810</resolution>
34      <noise>low</noise>
35      <blurredness>231.787</blurredness>
36    </optical>
37    <geometrical>
38      <distortion>
39        <sx>0.922018348623853</sx>
40        <sy>0.686006825938567</sy>
41        <rx>-0.038655462184874</rx>
42        <ry>0</ry>
43        <tx>0</tx>
44        <ty>92</ty>
45        <px>-3.27653997378768e-05</px>
46        <py>0</py>
47      </distortion>

```

E. Auflistung von Dateiinhalten

```
48      <rotation>2.00934289847729</rotation>
49      <arrangement>horizontal</arrangement>
50      <cover orientation="vertical">5</cover>
51  </geometrical>
52  <typographical>
53      <font>standard</font>
54      <language>german</language>
55  </typographical>
56  <difficult>false</difficult>
57  <polygon>
58      <username>pixtract</username>
59      <pt>
60          <x>667</x>
61          <y>905</y>
62      </pt>
63      <pt>
64          <x>3047</x>
65          <y>905</y>
66      </pt>
67      <pt>
68          <x>3047</x>
69          <y>1198</y>
70      </pt>
71      <pt>
72          <x>667</x>
73          <y>1198</y>
74      </pt>
75  </polygon>
76  </object>
77 </annotation>
```

Listing E.4: XML-Annotation für Abbildung 8.2 auf Seite 214.

ANHANG F

BEISPIELERGEBNISSE FÜR DEN GIST-BASIERTEN FILTER

In Abschnitt 7.4 wurde ein Vorgehen zur Ermittlung von Ausreißern innerhalb einer Menge von Bildern einer gegebenen Kategorie beschrieben. In diesem Abschnitt werden die Ausgaben des GIST-basierten Filters anhand einiger Beispielkategorien aus ImageNet vorgestellt. Die Beispiele wurden dabei nach ihren Eigenschaften in die Gruppen starre Objekte in Abschnitt F.1, bewegliche und verformbare Objekte in Abschnitt F.2, Objekte ohne festgelegter Form Abschnitt F.3 sowie Szenen Abschnitt F.4 eingeteilt.

F.1 Starre Objekte

Unter starren Objekten wurde diejenigen Kategorien eingesortiert, bei denen die abgebildeten Objekte eine prägende Erscheinungsform haben. Es können dabei jedoch unterschiedliche Formen durch verschiedene Blickwinkel oder Schnitte entstehen.

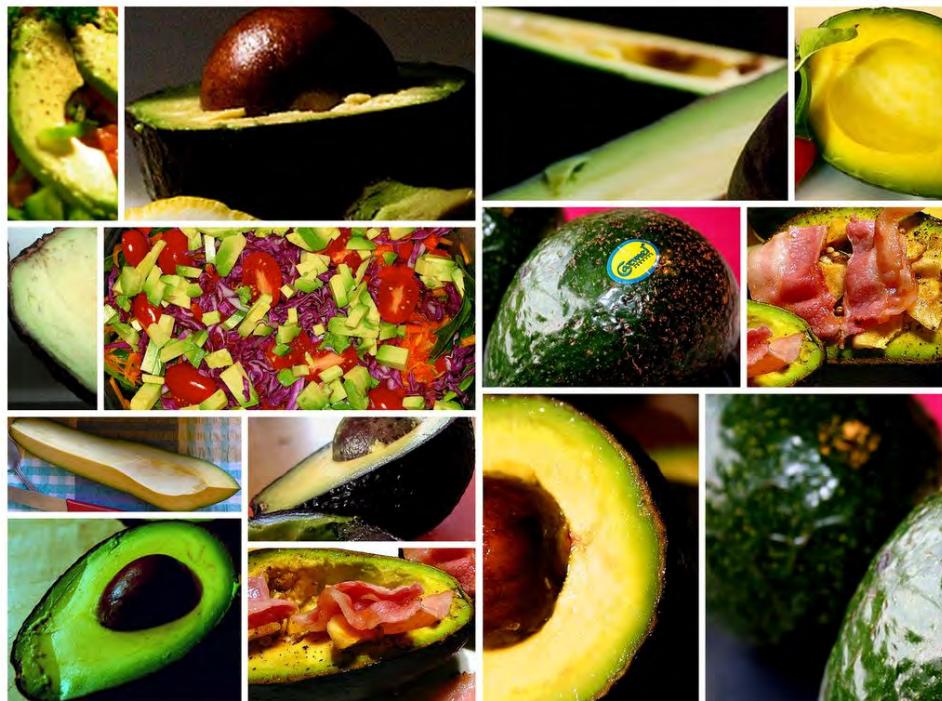
Die Trennung der guten und schlechten Bilder einer Kategorie ist für starre Objekte deutlich wahrnehmbar. Ausreißer sind in den meisten Fällen Bilder in denen nur ein Ausschnitt des Objekts zu sehen ist, teilweise auch durch Tiefenunschärfe verschwommen, Abdeckungen durch Personen oder andere Gegenstände, gravierende farbliche Abwei-

chungen, sowie extreme Beleuchtungseffekte (wie z. B. Silhouette eines Objekts). Es werden also genau diejenigen Bilder als Ausreißer markiert, welche bei der Definition der Anforderungen an den Filtermechanismus in Abschnitt 7.4.1 gefordert wurden.

Interessant sind unter Anderem die Ergebnisse für die Kategorie „Fernseher“ in Abbildung F.7, wo die Ausreißer aus Bildern bestehen, bei denen der Fernseher eingeschaltet ist. Da auf den Bildschirmen unterschiedliche Programme zu sehen sind, ist die Trennung durch den GIST-basierten Filter nachvollziehbar. Ähnlich interessant ist auch die Trennung bei der Kategorie „Tennisball“ in Abbildung F.8, da hier Objekte mit einem Schriftzug als Ausreißer eingestuft werden. Generell ist die Auswahl der besten Bilder bzw. der Ausreißer durch den Filter sehr gut.



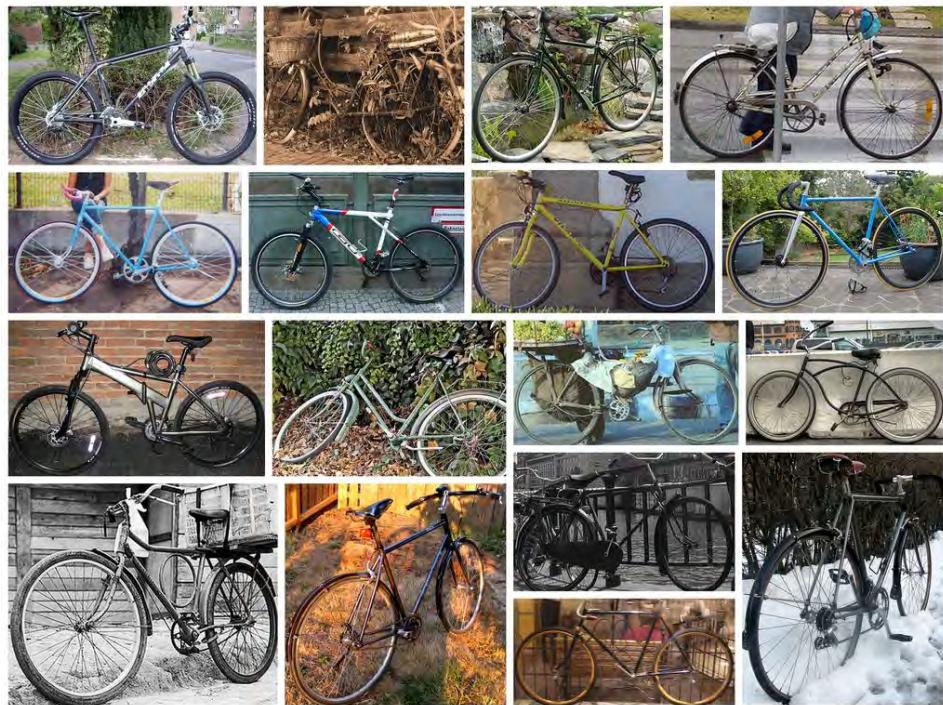
(a) beste Bilder



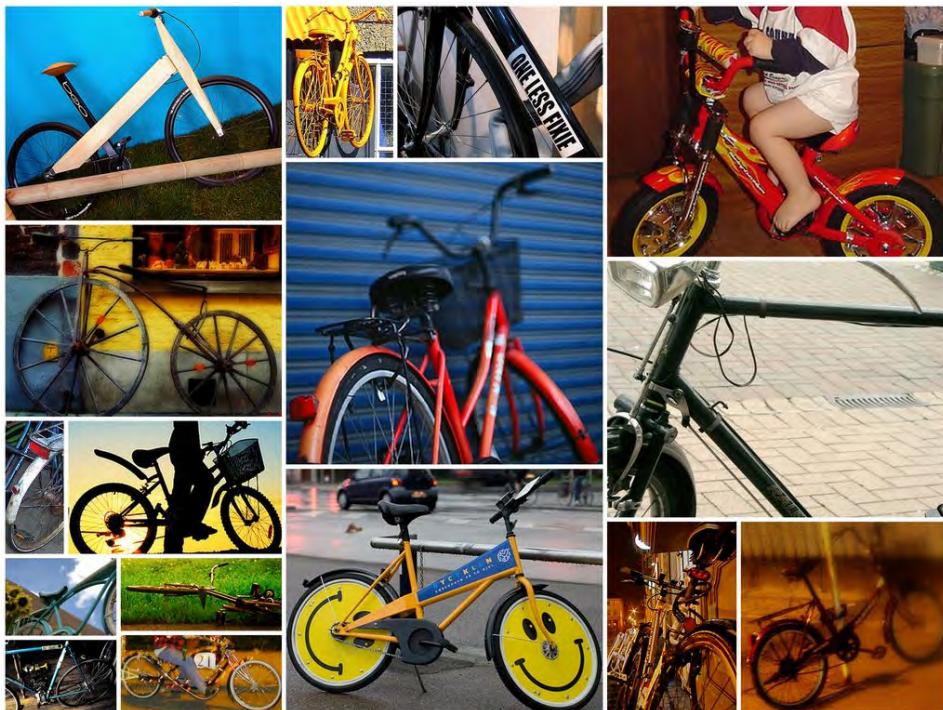
(b) schlechteste Bilder

Bild F.1: Beispiel für beste und schlechteste Bilder für die Kategorie „avocado, alligator pear, avocado pear, aguacate“ (WordNetID: 07764847).

F. Beispieldaten für den GIST-basierten Filter

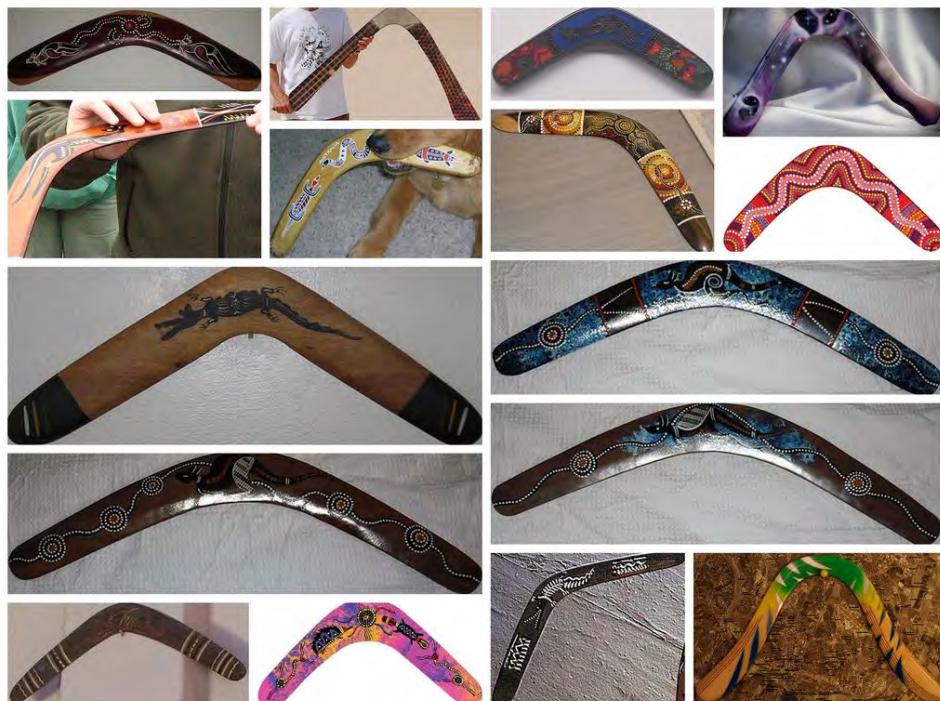


(a) beste Bilder

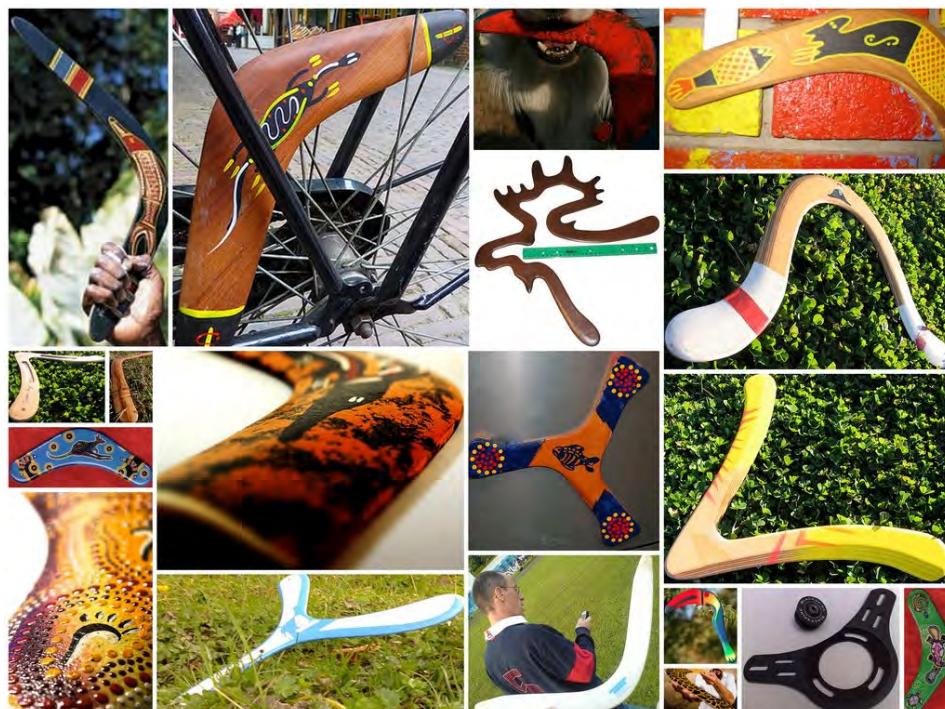


(b) schlechteste Bilder

Bild F.2: Beispiel für beste und schlechteste Bilder für die Kategorie „bicycle, bike, wheel, cycle“ (WordNetID: 02834778).



(a) beste Bilder



(b) schlechteste Bilder

Bild F.3: Beispiel für beste und schlechteste Bilder für die Kategorie „boomerang, throwing stick, throw stick“ (WordNetID: 02871963).

F. Beispieldaten für den GIST-basierten Filter



(a) beste Bilder



(b) schlechteste Bilder

Bild F.4: Beispiel für beste und schlechteste Bilder für die Kategorie „brake“ (WordNetID: 02889425).



(a) beste Bilder



(b) schlechteste Bilder

Bild F.5: Beispiel für beste und schlechteste Bilder für die Kategorie „lemon“ (WordNetID: 07749582).

F. Beispieldergebnisse für den GIST-basierten Filter



(a) beste Bilder



(b) schlechteste Bilder

Bild F.6: Beispiel für beste und schlechteste Bilder für die Kategorie „telephone, phone, telephone set“ (WordNetID: 04401088).



(a) beste Bilder



(b) schlechteste Bilder

Bild F.7: Beispiel für beste und schlechteste Bilder für die Kategorie „television, television system“ (WordNetID: 04404412).

F. Beispielergebnisse für den GIST-basierten Filter



(a) beste Bilder



(b) schlechteste Bilder

Bild F.8: Beispiel für beste und schlechteste Bilder für die Kategorie „tennis ball“ (WordNetID: 04409515).

F.2 Bewegliche und verformbare Objekte

In die Gruppe der beweglichen und verformbaren Objekte wurden diejenigen Gegenstände und Lebewesen eingeteilt, welche von sich aus in stark variablen Haltungen und Formen auftreten können. Die Ausreißer unterscheiden sich auch hier deutlich von den besten Bildern der Kategorie. Sie sind häufig durch Abdeckungen, Ausschnitte, sowie extremer Muster geprägt, erfüllen also auch für diese Gruppe von Objekten die Anforderungen an den Filter aus Abschnitt 7.4.1.

F. Beispieldergebnisse für den GIST-basierten Filter



(a) beste Bilder



(b) schlechteste Bilder

Bild F.9: Beispiel für beste und schlechteste Bilder für die Kategorie „kuvasz“ (WordNetID: 02104029).



(a) beste Bilder



(b) schlechteste Bilder

Bild F.10: Beispiel für beste und schlechteste Bilder für die Kategorie „paper“ (WordNetID: 06255613).

F. Beispieldergebnisse für den GIST-basierten Filter



(a) beste Bilder



(b) schlechteste Bilder

Bild F.11: Beispiel für beste und schlechteste Bilder für die Kategorie „towel“ (WordNetID: 04459362).

F.3 Objekte ohne fester Form

Eine besondere Schwierigkeit ist bei der Erkennung von Objekten ohne einer festen oder üblichen Form gegeben, da hier prägende Merkmale nur bedingt extrahierbar sind. Teilweise können jedoch auch bei diesen Objekten einige Ausreißer identifiziert werden, z. B. durch Verpackungen oder Tropfeffekte bei der Kategorie „Milch“ in Abbildung F.14. Bei den meisten Kategorien ist jedoch kaum ein Unterschied zwischen den besten Bildern und den Ausreißern feststellbar, da die Variabilität innerhalb einer Kategorie für diese Gruppe von Objekten sehr groß ist.

F. Beispieldergebnisse für den GIST-basierten Filter



(a) beste Bilder

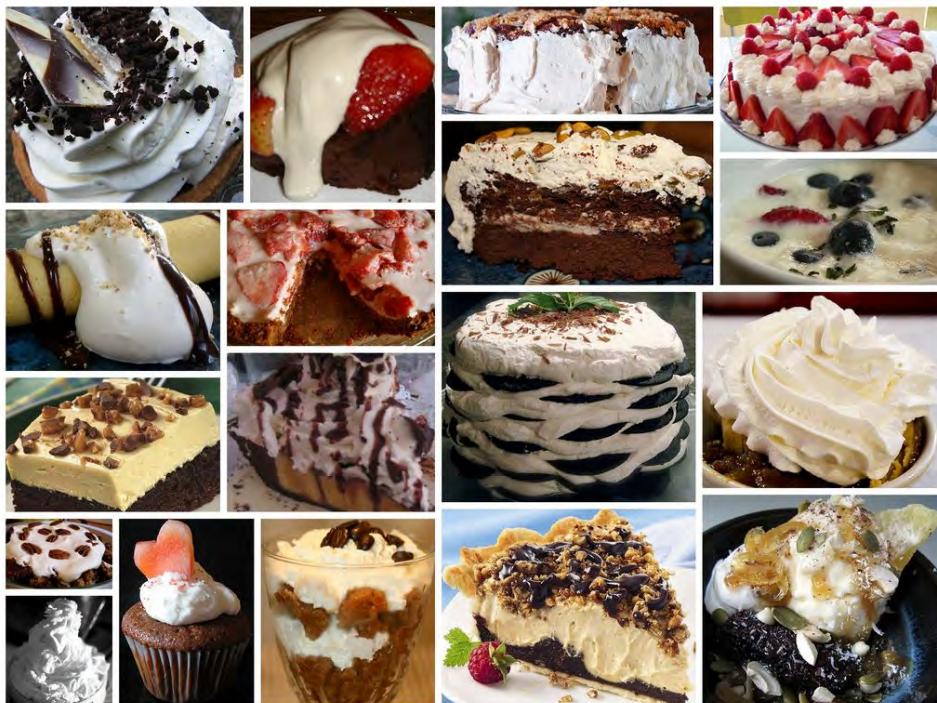


(b) schlechteste Bilder

Bild F.12: Beispiel für beste und schlechteste Bilder für die Kategorie „butter“ (WordNetID: 07848338).



(a) beste Bilder



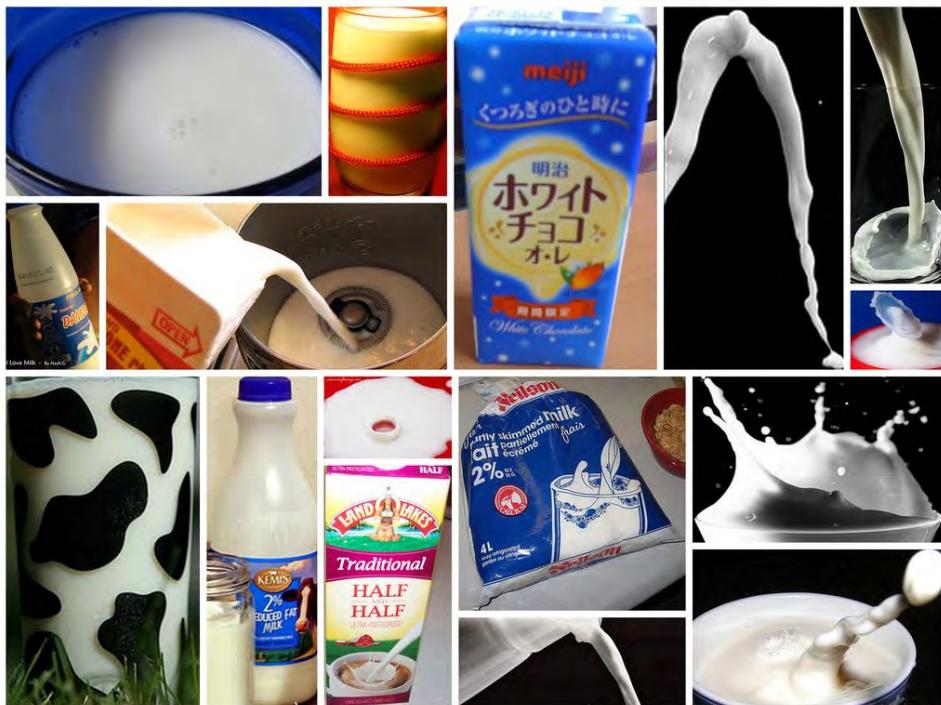
(b) schlechteste Bilder

Bild F.13: Beispiel für beste und schlechteste Bilder für die Kategorie „double creme, heavy whipping creme“ (WordNetID: 07847585).

F. Beispieldergebnisse für den GIST-basierten Filter

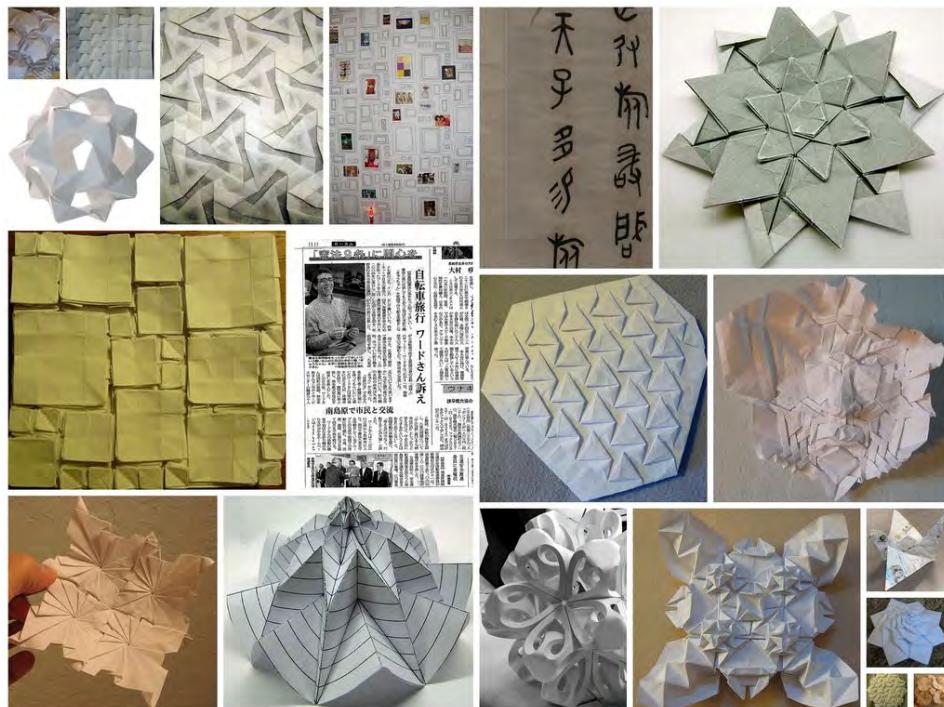


(a) beste Bilder



(b) schlechteste Bilder

Bild F.14: Beispiel für beste und schlechteste Bilder für die Kategorie „milk“ (WordNetID: 07844042).



(a) beste Bilder



(b) schlechteste Bilder

Bild F.15: Beispiel für beste und schlechteste Bilder für die Kategorie „paper“ (WordNetID: 14974264).

F.4 Szenen

ImageNet beinhaltet neben Objekten auch Bilder zu verschiedenen Szenen. Die Trennung zwischen den besten Bildern und Ausreißern ist auch bei dieser Gruppe von Kategorien zum Teil gut sichtbar. Bei der Kategorie „Billard-Saal“ in Abbildung F.16 werden z. B. Aufnahmen von Billardtischen klar als Ausreißer markiert. Schlechter schneiden jedoch Kategorien mit extrem hoher Diversität ab, wie z. B. bei der Kategorie „Biss, Snack“ in Abbildung F.17 oder bei „Pasta“ in Abbildung F.19.



(a) beste Bilder



(b) schlechteste Bilder

Bild F.16: Beispiel für beste und schlechteste Bilder für die Kategorie „billiard room, billiard saloon, billiard parlor, billiard parlour, billiard hall“ (WordNetID: 02839592).

Danksagung



(a) beste Bilder



(b) schlechteste Bilder

Bild F.17: Beispiel für beste und schlechteste Bilder für die Kategorie „bite, collation, snack“ (WordNetID: 07577374).



(a) beste Bilder



(b) schlechteste Bilder

Bild F.18: Beispiel für beste und schlechteste Bilder für die Kategorie „open-air market, open-air marketplace, market square“ (WordNetID: 03847823).

Danksagung



(a) beste Bilder



(b) schlechteste Bilder

Bild F.19: Beispiel für beste und schlechteste Bilder für die Kategorie „pasta“ (WordNetID: 07863374).

DANKSAGUNG

An dieser Stelle möchte ich allen danken, die zum Gelingen dieser Arbeit beigetragen haben.

Meinem Doktorvater und Erstgutachter Herrn Prof. Dr.-Ing. Klaus Meyer-Wegener danke ich für die Betreuung meiner Arbeit. Trotz vieler Ämter und Aufgaben unterstützte er mich in fachlichen Diskussionen, insbesondere bei der Eingrenzung meines Dissertationsthemas sowie in der Schlussphase meiner Arbeit. Auch über privaten Angelegenheiten konnten wir uns stets gut verständigen. Ganz herzlich danke ich meinem Zweitgutachter Herrn Prof. Dr. Harald Kosch. Sowohl bei den regelmäßigen MMIS-Treffen, als auch bei der ACM Multimedia Konferenz in Florenz sowie in der Endphase meiner Dissertation half er mir durch rege Unterstützung bei meiner Arbeit. Herrn Prof. Dr.-Ing. André Kaup danke ich für die Rolle des fachfremden Prüfers. Herrn Prof. Dr.-Ing. Richard Lenz danke ich für die Übernahme des Prüfungsvorsitzes. Seit dem ersten Tag waren wir in enger Verbindung durch die gemeinsame Betreuung einer großen Lehrveranstaltung.

Ein ganz besonderer Dank gilt meinen Kollegen. Meinem langjährigen Zimmernachbarn Herrn Dr.-Ing. Michael Daum danke ich für die stets offene Tür. Bei jetweder Fragestellung stand er mir immer mit Rat zur Seite. Auch gilt ihm mein Dank für den Namen meines Projekts. Meiner zweiten langjährigen Zimmernachbarin Frau Ursula Büttner danke ich ebenfalls für ein stets offenes Ohr sowie für die zahlreichen Tipps für alle Lebenslagen. Herrn Dr.-Ing. Udo Mayer danke ich für den initialen Crash-Kurs für die Lehre direkt am Anfang meiner Lehrstuhlzeit. In diesem 15 minütigen Vortrags-Training habe ich weit mehr gelernt, als in allen anschließenden Fortbildungsseminaren. Herrn Dipl.-Inf. Thomas Fischer danke ich für die schöne Zeit in Budapest sowie für die gemeinsamen kulinarischen Expeditionen und Kochevents. Tom, ich hoffe Du wirst die Fahne des

Danksagung

kulinarischen Anspruchs am Lehrstuhl weiterhin oben halten und die Tradition weiterführen! Frau Brigitte Knechtel danke ich für die schnelle und problemlose Abwicklung von diversen organisatorischen Angelegenheiten sowie für die zahlreichen kulinarisch wertvollen Unterhaltungen. Herrn Dipl.-Inf. Frank Lauterwald danke ich für seinen Sinn für Humor und Ironie sowie seiner Kameratasche, die mich auf Konferenzen immer begleitet hat. Frau Dr.-Ing. Juliane Blechinger danke ich für die stets aufmunternden Gespräche auf meinem Bürossofa. Sie hat mir immer ein Lächeln geschenkt, mit dem der Tag dann viel leichter zu meistern war. Mein Dank gilt Herrn Dr.-Ing. Florian Irmert für das Durchhaltevermögen an den langen Tagen bei der Abwicklung unserer gemeinsam betreuten Praktika sowie für die vielen leckeren Cocktail-Tipps. Herrn Dipl.-Inf. Johannes Held danke ich für die anregenden Diskussionen über Musik sowie für die vegetarischen und veganen Genüsse. Meinem Zimmernachbarn in der zweiten Hälfte meiner Zeit am Lehrstuhl Herrn Dipl.-Inf. Christoph Neumann danke ich für die erfolgreiche Betreuung unserer gemeinsamen Lehrveranstaltung. Ich bin jedoch weiterhin eher für die Keule als für die Zauberstäbe. Herrn Dipl.-Inf. Julian Rith danke ich für die interessanten Gespräche über neue Webtechniken und Fotografie. Ich freue mich, dass bei ihm mein HDR-Tutorial in guten Händen aufgehoben ist. Bei Herrn Dipl.-Inf. Gregor Endler bedanke ich mich für die schönen Grillabende. Herrn Dipl.-Inf. Peter Schwab danke ich für die erfolgreiche Übernahme meiner langjährigen Lehrveranstaltung und meiner lehrstuhlinternen Aufgaben. Herrn Dipl.-Inf. Philipp Baumgärtel und Herrn Dipl.-Ing. Niko Pollner danke ich für die Unterstützung bei der Abwicklung unserer gemeinsamen Lehrveranstaltung. Für die Diskussionen über Audioverarbeitung sowie insbesondere für die leckeren polnischen gastronomischen Genüsse danke ich Herrn Dr.-Ing. Maciej Suchomski. Herrn Dr.-Ing. Vladimir Entin danke ich für die wunderschöne Zeit in seiner neuen Heimat in Vorarlberg. Herrn Dr.-Ing. Henning Weiler danke ich zum Austausch über Flächen, Sphären und Soundscapes. Herrn Dr.-Ing. Marcus Meyerhöfer und Herrn Prof. Dr.-Ing. Sascha Müller-Feuerstein danke ich für ihre Unterstützung durch ihre langjährigen Erfahrungen. Frau Nadezda Jelani danke ich für die Abwicklung von organisatorischen Angelegenheiten. Frau Roswitha Braun und Frau Ursula Stoyan danke ich für die schnelle und unkomplizierte technische Hilfe.

Bedanken möchte ich mich auch bei meinen ehemaligen Studenten, ohne deren Hilfe diese Dissertation nicht vollendet werden hätte können: Dipl.-Inf. Giacomo Inches, Dipl.-Ing. Andrei Galea, Dipl.-Inf. Anders Dicker, Dipl.-Inf. Julian Rith, Dipl.-Ing. Alexander Uhl, M. Sc. Jun Chen und Sergiy Protsenko. Des Weiteren gilt mein Dank auch Herrn Dipl.-Inf. Christian Riess, der mich am Anfang durch fachliche Diskussionen bei der

Themenfindung unterstützt hat. Meinem Landsmann M. Sc. Attila Budai danke ich für die medizinischen Aufnahmen, dessen Verwendung er mir für meine Dissertation freundlicherweise überlassen hat.

Abschließend möchte ich mich bei meiner Familie bedanken. Meiner Schwester Andrea Adelhardt und meinem Schwager Daniel Adelhardt danke ich für die Wegbereitung für mein Studium in Erlangen sowie für die entspannenden Wochenenden und Ausflüge in Oberbayern. Mein Dank gilt auch Hans und Gundi Adelhardt. Sie standen mir mit ihrer Hilfe vor allem in meiner Erlanger Anfangszeit zur Seite. Auch danke ich ihnen für die schönen Wanderungen in der wundervollen Fränkischen Schweiz. Mein ganz besonderer Dank gilt meinen Eltern János und Gudrun Nagy. Es war kein einfacher Schritt mit 18 Jahren weit weg in eine vollkommen neue Umgebung zu ziehen. Ich danke meinen Eltern für den Rückhalt sowohl während meines Studiums, als auch während meiner Zeit am Lehrstuhl. Vor allem danke ich meinen Eltern auch für die uneingeschränkte Unterstützung bei meinen zahlreichen privaten Projekten. Es werden in der Zukunft sicherlich noch viele hinzu kommen.

LITERATURVERZEICHNIS

- [AD04] AHN, Luis von ; DABBISH, Laura: Labeling Images with a Computer Game. In: *ACM SIGCHI Conference on Human Factors in Computing Systems*, 2004, S. 319–326. <http://dx.doi.org/10.1145/985692.985733>
- [AE97] ARMITAGE, Linda H. ; ENSER, Peter G.: Analysis of user need in image archives. In: *Journal of Information Science* 23 (1997), Nr. 4, S. 287–299. <http://dx.doi.org/10.1177/016555159702300403>
- [AF10] AMATO, Giuseppe ; FALCHI, Fabrizio: kNN Based Image Classification Relying on Local Feature Similarity. In: *ACM International Conference on Similarity Search and Applications*, 2010, S. 101–108. <http://dx.doi.org/10.1145/1862344.1862360>
- [AFB10] AMATO, Giuseppe ; FALCHI, Fabrizio ; BOLETTIERI, Paolo: Recognizing Landmarks Using Automated Classification Techniques: an Evaluation of Various Visual Features. In: *IEEE International Conference on Advances in Multimedia*, 2010, S. 78–83. <http://dx.doi.org/10.1109/MMEDIA.2010.20>
- [AG87] APOSTOLICO, A. ; GUERRA, C.: The longest common subsequence problem revisited. In: *Algorithmica* 2 (1987), Nr. 1-4, S. 315–336. <http://dx.doi.org/10.1007/BF01840365>
- [AGK⁺06] AHN, Luis von ; GINOSAR, Shiry ; KEDIA, Mihir ; LIU, Ruoran ; BLUM, Manuel: Improving accessibility of the web with a computer game. In: *ACM SIGCHI Conference on Human Factors in Computing Systems*, 2006, S. 79–82. <http://dx.doi.org/10.1145/1124772.1124785>

- [AI08] ANDONI, Alexandr ; INDYK, Piotr: Near-Optimal Hashing Algorithms for Approximate Nearest Neighbor in High Dimensions. In: *Communications of the ACM* 51 (2008), Nr. 1, S. 117–122. <http://dx.doi.org/10.1145/1327452.1327494>
- [AMFM09] ARBELÁEZ, Pablo ; MAIRE, Michael ; FOWLKES, Charless ; MALIK, Jitendra: From contours to regions: An empirical evaluation. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2009, S. 2294–2301. <http://dx.doi.org/10.1109/CVPR.2009.5206707>
- [Ass11] ASSOCATION, Camera & Imaging P.: *Production, Shipment of Digital Still Camera*. http://www.cipa.jp/english/data/pdf/d-201112_e.pdf. Version: 2011
- [AYV07] AKBAS, Emre ; YARMAN VURAL, Fatos T.: Automatic Image Annotation by Ensemble of Visual Descriptors. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2007, S. 1–8. <http://dx.doi.org/10.1109/CVPR.2007.383484>
- [BAG03] BERRANI, Sid-Ahmed ; AMSALEG, Laurent ; GROS, Patrick: Approximate Searches: k-Neighbors + Precision. In: *ACM International Conference on Information and Knowledge Management*, 2003, S. 24–31. <http://dx.doi.org/10.1145/956863.956870>
- [BBE03] BURFORD, Bryan ; BRIGGS, Pam ; EAKINS, John P.: A Taxonomy of the Image: On the Classification of Content for Image Retrieval. In: *Visual Communication* 2 (2003), Nr. 2, S. 123–161. <http://dx.doi.org/10.1177/1470357203002002001>
- [BBK⁺00] BERCHTOLD, Stefan ; BÖHM, Christian ; KEIM, Daniel A. ; KRIEGEL, Hans-Peter ; XU, Xiaowei: Optimal Multidimensional Query Processing Using Tree Striping. In: *International Conference on Data Warehousing and Knowledge Discovery*, 2000, S. 244–257. http://dx.doi.org/10.1007/3-540-44466-1_24
- [BBK01] BÖHM, Christian ; BERCHTOLD, Stefan ; KEIM, Daniel A.: Searching in High-Dimensional Spaces – Index Structures for Improving the Performance of Multimedia Databases. In: *ACM Computing Surveys* 33 (2001), Nr. 3, S. 322–373. <http://dx.doi.org/10.1145/502807.502809>

- [BBM05] BERG, Alexander C. ; BERG, Tamara L. ; MALIK, Jitendra: Shape matching and object recognition using low distortion correspondences. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2005, S. 26–33. <http://dx.doi.org/10.1109/CVPR.2005.320>
- [BCB⁺07] BIGHAM, Jeffrey P. ; CAVENDER, Anna C. ; BRUDVIK, Jeremy T. ; WOBBROCK, Jacob O. ; LADNER, Richard E.: WebinSitu: a comparative analysis of blind and sighted browsing behavior. In: *ACM SIGACCESS International Conference on Computers and Accessibility*, 2007, S. 51–58. <http://dx.doi.org/10.1145/1296843.1296854>
- [BDL10] BERG, Alexander C. ; DENG, Jia ; LI, Fei-Fei: *ImageNet Large Scale Visual Recognition Challenge 2010*. <http://www.image-net.org/challenges/LSVRC/2010/>. Version: 2010
- [BDL11] BERG, Alexander C. ; DENG, Jia ; LI, Fei-Fei: *ImageNet Large Scale Visual Recognition Challenge 2011*. <http://www.image-net.org/challenges/LSVRC/2011/>. Version: 2011
- [BDPDPM93] BROWN, Peter F. ; DELLA PIETRA, Vincent J. ; DELLA PIETRA, Stephen A. ; MERCER, Robert L.: The Mathematics of Statistical Machine Translation: Parameter Estimation. In: *Computational Linguistics* 19 (1993), Nr. 2, S. 263–311
- [Ber05] BERTRAM, Bernd: Blindheit und Sehbehinderung in Deutschland: Ursachen und Häufigkeit. In: *Der Augenarzt* 39 (2005), Nr. 6, S. 267–268
- [BETVG08] BAY, Herbert ; ESS, Andreas ; TUYTELAARS, Tinne ; VAN GOOL, Luc: Speeded-Up Robust Features (SURF). In: *Computer Vision and Image Understanding* 110 (2008), Nr. 3, S. 346–359. <http://dx.doi.org/10.1016/j.cviu.2007.09.014>
- [BF01] BARNARD, Kobus ; FORSYTH, David: Learning the semantics of words and pictures. In: *IEEE International Conference on Computer Vision* Bd. 2, 2001, S. 408–415. <http://dx.doi.org/10.1109/ICCV.2001.937654>
- [BGJT04] BLEI, David M. ; GRIFFITHS, Thomas L. ; JORDAN, Michael I. ; TENENBAUM, Joshua B.: Hierarchical Topic Models and the Nested Chinese Restaurant Process. In: *Advances in Neural Information Processing Systems*, 2004, S. 17–24

- [BGRS99] BEYER, Kevin ; GOLDSTEIN, Jonathan ; RAMAKRISHNAN, Raghu ; SHAFT, Uri: When Is "Nearest Neighbor" Meaningful? In: *International Conference on Database Theory*, 1999, S. 217–235. http://dx.doi.org/10.1007/3-540-49257-7_15
- [BH04] BRATAAS, Gunnar ; HUGHES, Peter: Exploring Architectural Scalability. In: *International Workshop on Software and Performance*, 2004, S. 125–129. <http://dx.doi.org/10.1145/974043.974064>
- [BHR00] BERGROTH, L. ; HAKONEN, H. ; RAITA, T.: A survey of longest common subsequence algorithms. In: *International Symposium on String Processing and Information Retrieval*, 2000, S. 39–48. <http://dx.doi.org/10.1109/SPIRE.2000.878178>
- [Bie87] BIEDERMAN, Irving: Recognition-by-Components: A Theory of Human Image Understanding. In: *Psychological Review* 94 (1987), Nr. 2, S. 115–147
- [Bil06] BILESCHEI, Stanley M.: *StreetScenes: Towards Scene Understanding in Still Images*, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Dissertation, 2006
- [Bim99] BIMBO, Alberto del ; BIMBO, Alberto del (Hrsg.): *Visual Information Retrieval*. Morgan Kaufmann, 1999
- [BKL⁺06] BIGHAM, Jeffrey P. ; KAMINSKY, Ryan S. ; LADNER, Richard E. ; DANIELSSON, Oscar M. ; HEMPTON, Gordon L.: WebInSight: Making Web Images Accessible. In: *ACM SIGACCESS International Conference on Computers and Accessibility*, 2006, S. 181–188. <http://dx.doi.org/10.1145/1168987.1169018>
- [BLJ04] BACH, Francis R. ; LANCKRIET, Gert R. G. ; JORDAN, Michael I.: Multiple Kernel Learning, Conic Duality, and the SMO Algorithm. In: *International Conference on Machine Learning*, 2004, S. 6. <http://dx.doi.org/10.1145/1015330.1015424>
- [BLK⁺09] BIZER, Christian ; LEHMANN, Jens ; KOBILAROV, Georgi ; AUER, Sören ; BECKER, Christian ; CYGANIAK, Richard ; HELLMANN, Sebastian: DBpedia - A Crystallization Point for the Web of Data. In: *Journal of Web Semantics: Science, Services and Agents on the World Wide Web* 7 (2009), Nr. 3, S. 154–165. <http://dx.doi.org/10.1016/j.websem.2009.06.001>

2009.07.002

- [BM98] BOYAN, Justin A. ; MOORE, Andrew W.: Learning Evaluation Functions for Global Optimization and Boolean Satisfiability. In: *AAAI International Joint Conference on Artificial Intelligence*, 1998, S. 3–10
- [BMM07] BOSCH, Anna ; MUÑOZ, Xavier ; MARTÍ, Robert: A review: Which is the best way to organize/classify images by content? In: *Image and Vision Computing* 25 (2007), Nr. 6, S. 778–791. <http://dx.doi.org/10.1016/j.imavis.2006.07.015>
- [BMP02] BELONGIE, Serge ; MALIK, Jitendra ; PUZICHA, Jan: Shape Matching and Object Recognition Using Shape Contexts. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (2002), Nr. 4, S. 509–522. <http://dx.doi.org/10.1109/34.993558>
- [Bon00] BONDI, André B.: Characteristics of Scalability and their Impact on Performance. In: *International Workshop on Software and Performance*, 2000, S. 195–203. <http://dx.doi.org/10.1145/350391.350432>
- [BPPW08] BART, Evgeniy ; PORTEOUS, Ian ; PERONA, Pietro ; WELLING, Max: Unsupervised Learning of Visual Taxonomies. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2008, S. 1–8. <http://dx.doi.org/10.1109/CVPR.2008.4587620>
- [BR01] BROWN, Silas S. ; ROBINSON, Peter: A World Wide Web Mediator for Users with Low Vision. In: *Conference on Human Factors in Computing Systems Workshop No. 14.*, 2001
- [BR07] BOSCH RUÉ, Anna: *Image classification for large number of object categories*, Department of Electronics, Informatics and Automation, University of Girona, Dissertation, 2007
- [BSI08] BOIMAN, Oren ; SCHECHTMAN, Eli ; IRANI, Michal: In Defense of Nearest-Neighbor Based image Classification. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2008, S. 1063–1071. <http://dx.doi.org/10.1109/CVPR.2008.4587598>
- [Bud29] BUDGE, Ernest A. W.: *The Rosetta Stone in the British Museum*. Religious Tract Society, London, 1929

- [BVBF07] BLANKEN, Henk ; VRIES, Arjen P. ; BLOK, Henk E. ; FENG, Ling: *Multimedia Retrieval*. Springer, 2007. <http://dx.doi.org/10.1007/978-3-540-72895-5>
- [BZM07] BOSCH, Anna ; ZISSERMAN, Andrew ; MUÑOZ, Xavier: Representing shape with a spatial pyramid kernel. In: *ACM International Conference on Image and Video Retrieval*, 2007, S. 401–408. <http://dx.doi.org/10.1145/1282280.1282340>
- [BZM08] BOSCH, Anna ; ZISSERMAN, Andrew ; MUÑOZ, Xavier: Scene Classification Using a Hybrid Generative/Discriminative Approach. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (2008), Nr. 4, S. 712–727. <http://dx.doi.org/10.1109/TPAMI.2007.70716>
- [CAG00] CONNISS, Lynne R. ; ASHFORD, Julie A. ; GRAHAM, Margaret E.: Information Seeking Behaviour in Image Retrieval: VISOR I Final Report / University of Northumbria in Newcastle, Institute for Image and Data Research. 2000. – Forschungsbericht
- [Car04] CARNEIRO, Gustavo Henrique Monteiro De B.: *Image pattern recognition using phase-based local features and their flexible spatial configuration*, University of Toronto, Dissertation, 2004
- [CBGZ06] CESA-BIANCHI, Nicoló ; GENTILE, Claudio ; ZANIBONI, Luca: Incremental Algorithms for Hierarchical Classification. In: *Journal of Machine Learning Research* 7 (2006), Nr. 1, S. 31–54
- [CBV09] CAMPOS, Teófilo E. ; BABU, Modla R. ; VARMA, Manik: Character Recognition in Natural Images. In: *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2009, S. 273–280
- [CCG04] CHEN, Yangchi ; CRAWFORD, Melba M. ; GHOSH, Joydeep: Integrating Support Vector Machines in a Hierarchical Output Space Decomposition Framework. In: *IEEE International Geoscience and Remote Sensing Symposium* Bd. 2, 2004, S. 949–952. <http://dx.doi.org/10.1109/IGARSS.2004.1368565>
- [CCMV07] CARNEIRO, Gustavo ; CHAN, Antoni B. ; MORENO, Pedro J. ; VASCONCELOS, Nuno: Supervised Learning of Semantic Classes for Image Annotation and Retrieval. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29 (2007), Nr. 3, S. 394–410. <http://dx.doi.org/10.1109/TPAMI.2007.70716>

- //dx.doi.org/10.1109/TPAMI.2007.61
- [CDG⁺06] CHANG, Fay ; DEAN, Jeffrey ; GHEMAWAT, Sanjay ; HSIEH, Wilson C. ; WALLACH, Deborah A. ; BURROWS, Mike ; CHANDRA, Tushar ; FIKES, Andrew ; GRUBER, Robert E.: Bigtable: A Distributed Storage System for Structured Data. In: *Symposium on Operating Systems Design and Implementation (OSDI)*, 2006, S. 205–218
- [CDG⁺08] CHANG, Fay ; DEAN, Jeffrey ; GHEMAWAT, Sanjay ; HSIEH, Wilson C. ; WALLACH, Deborah A. ; BURROWS, Mike ; CHANDRA, Tushar ; FIKES, Andrew ; GRUBER, Robert E.: Bigtable: A Distributed Storage System for Structured Data. In: *ACM Transactions on Computer Systems* 26 (2008), Nr. 2, S. 4. <http://dx.doi.org/10.1145/1365815.1365816>
- [Che01] CHEN, Hsin-liang: An Analysis of Image Queries in the Field of Art History. In: *Journal of the American Society for Information Science and Technology* 52 (2001), Nr. 3, S. 260–273. [http://dx.doi.org/10.1002/1532-2890\(2000\)9999:9999<::AID-ASI1606>3.0.CO;2-M](http://dx.doi.org/10.1002/1532-2890(2000)9999:9999<::AID-ASI1606>3.0.CO;2-M)
- [CHH07] CAI, Deng ; HE, Xiaofei ; HAN, Jiawei: Efficient Kernel Discriminant Analysis via Spectral Regression. In: *IEEE International Conference on Data Mining*, 2007, S. 427–432. <http://dx.doi.org/10.1109/ICDM.2007.88>
- [Cho10] CHOI, Youngok: Investigating Variation in Querying Behavior for Image Searches on the Web. In: *Proceedings of the American Society for Information Science and Technology* Bd. 47, 2010, S. 1–10. <http://dx.doi.org/10.1002/meet.14504701220>
- [CJ03] CARNEIRO, G. ; JEPSON, A.D.: Multi-scale phase-based local features. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 1, 2003, S. 736–743. <http://dx.doi.org/10.1109/CVPR.2003.1211426>
- [CJS05] CLOUGH, Paul ; JOHO, Hideo ; SANDERSON, Mark: Automatically Organising Images using Concept Hierarchies. In: *ACM SIGMM International Workshop on Multimedia Information Retrieval*, 2005
- [CKL⁺06] CHU, Cheng-Tao ; KIM, Sang K. ; LIN, Yi-An ; YU, YuanYuan ; BRADKSI, Gary ; NG, Andrew Y. ; OLUKOTUN, Kunle: Map-Reduce for Machine Learning on Multicore. In: *Advances in Neural Information Processing*

- Systems*, 2006, S. 281–288
- [CL06] CARNEIRO, Gustavo ; LOWE, David G.: Sparse Flexible Models of Local Features. In: *European Conference on Computer Vision*, 2006, S. 29–43. <http://dx.doi.org/10.1007/11744078>
- [CPR⁺07] CHIERICHETTI, Flavio ; PANCONESI, Alessandro ; RAGHAVAN, Prabhakar ; SOZIO, Mauro ; TIBERI, Alessandro ; UPFAL, Eli: Finding Near Neighbors Through Cluster Pruning. In: *ACM Symposium on Principles of Database Systems*, 2007, S. 103–112. <http://dx.doi.org/10.1145/1265530.1265545>
- [CR03] CHOI, Youngok ; RASMUSSEN, Edie M.: Searching for images: The analysis of users' queries for image retrieval in American history. In: *Journal of the American Society for Information Science and Technology* 54 (2003), Nr. 6, S. 498–511. <http://dx.doi.org/10.1002/asi.10237>
- [Cra06] CRAVEN, Timothy C.: Some features of alt texts associated with images in Web pages. In: *Information Research* 11 (2006), Nr. 2
- [CV95] CORTES, Corinna ; VAPNIK, Vladimir N.: Support-Vector Networks. In: *Machine Learning* Bd. 20, 1995, S. 273–297. <http://dx.doi.org/10.1023/A:1022627411411>
- [CW05] CASASENT, David ; WANG, Yu-Chiang: A hierarchical classifier using new support vector machines for automatic target recognition. In: *Neural Networks* 18 (2005), Nr. 5-6, S. 541–548. <http://dx.doi.org/10.1016/j.neunet.2005.06.033>
- [CZ09] CHANG, Loh Z. ; ZHIYING, Steven Z.: Robust Pre-processing Techniques for OCR Applications on Mobile Devices. In: *International Conference on Mobile Technology, Application & Systems*, 2009, S. 60. <http://dx.doi.org/10.1145/1710035.1710095>
- [CZJ07] CHAI, Joyce Y. ; ZHANG, Chen ; JIN, Rong: An Empirical Investigation of User Term Feedback in Text-Based Targeted Image Search. In: *ACM Transactions on Information Systems* 25 (2007), Nr. 1, S. 3. <http://dx.doi.org/10.1145/1198296.1198299>
- [Dam64] DAMERAU, Fred J.: A technique for computer detection and correction of spelling errors. In: *Communications of the ACM* 7 (1964), Nr. 3, S.

- 171–176. <http://dx.doi.org/10.1145/363958.363994>
- [DBFF02] DUYGULU, Pinar ; BARNARD, Kobus ; FREITAS, J. F. G. ; FORSYTH, D. A.: Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary. In: *European Conference on Computer Vision*, 2002, S. 97–112. http://dx.doi.org/10.1007/3-540-47979-1_7
- [DBLFF10] DENG, Jia ; BERG, Alexander C. ; LI, Kai ; FEI-FEI, Li: What Does Classifying More Than 10,000 Image Categories Tell Us? In: *European Conference on Computer Vision*, 2010, S. 71–84. http://dx.doi.org/10.1007/978-3-642-15555-0_6
- [DDS⁺⁰⁹] DENG, Jia ; DONG, Wei ; SOCHER, Richard ; LI, Li-Jia ; LI, Kai ; FEI-FEI, Li: ImageNet: A Large-Scale Hierarchical Image Database. In: *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, 2009, S. 248–255. <http://dx.doi.org/10.1109/CVPRW.2009.5206848>
- [DG04] DEAN, Jeffrey ; GHEMAWAT, Sanjay: MapReduce: Simplified Data Processing on Large Clusters. In: *Sixth Symposium on Operating System Design and Implementation*, 2004, S. 137–150
- [DG08] DEAN, Jeffrey ; GHEMAWAT, Sanjay: MapReduce: Simplified Data Processing on Large Clusters. In: *Communications of the ACM* 51 (2008), Nr. 1, S. 107–113. <http://dx.doi.org/10.1145/1327452.1327492>
- [DHS00] DUDA, Richard O. ; HART, Peter E. ; STORK, David G.: *Pattern Classification*. 2. Wiley & Sons, 2000
- [DI03] DORADO, A. ; IZQUIERDO, E.: Semi-automatic image annotation using frequent keyword mining. In: *IEEE International Conference on Information Visualization*, 2003, S. 532–535. <http://dx.doi.org/10.1109/IV.2003.1218036>
- [Dic11] DICKER, Anders: *Vergleich von OCR-Tools für die Texterkennung in Fotos*, Friedrich-Alexander-Universität Erlangen-Nürnberg, Diplomarbeit, 2011
- [DJLW08] DATTA, Ritendra ; JOSHI, Dhiraj ; LI, Jia ; WANG, James Z.: Image Retrieval: Ideas, Influences and Trends of the New Age. In: *ACM Computing Surveys* 40 (2008), Nr. 2, S. 5. <http://dx.doi.org/10.1145/1348246>.

1348248

- [DJS⁺09] DOUZE, Matthijs ; JÉGOU, Hervé ; SANDHAWALIA, Harsimrat ; AMSALEG, Laurent ; SCHMID, Cordelia: Evaluation of GIST descriptors for web-scale image search. In: *ACM International Conference on Image and Video Retrieval*, 2009, S. 19. <http://dx.doi.org/10.1145/1646396.1646421>
- [DKN04] DESELAERS, Thomas ; KEYSERS, Daniel ; NEY, Hermann: Classification error rate for quantitative evaluation of content-based image retrieval systems. In: *IEEE International Conference on Pattern Recognition* Bd. 2, 2004, S. 505–508. <http://dx.doi.org/10.1109/ICPR.2004.1334280>
- [DKN08] DESELAERS, Thomas ; KEYSERS, Daniel ; NEY, Hermann: Features for Image Retrieval: An Experimental Comparison. In: *Information Retrieval* 11 (2008), Nr. 2, S. 77–107. <http://dx.doi.org/10.1007/s10791-007-9039-3>
- [DLR77] DEMPSTER, A. P. ; LAIRD, N. M. ; RUBIN, D. B.: Maximum Likelihood from Incomplete Data via the EM Algorithm. In: *Journal of the Royal Statistical Society* 39 (1977), Nr. 1, S. 1–38
- [DLST09] DICKINSON, Sven J. (Hrsg.) ; LEONARDIS, Aleš (Hrsg.) ; SCHIELE, Bernt (Hrsg.) ; TARR, Michael J. (Hrsg.): *Object Categorization - Computer and Human Vision Perspectives*. Cambridge University Press, 2009
- [DQRJ⁺10] DITTRICH, Jens ; QUIANÉ-RUIZ, Jorge-Arnulfo ; JINDAL, Alekh ; KARGIN, Yagiz ; SETTY, Vinay ; SCHAD, Jörg: Hadoop++: Making a Yellow Elephant Run Like a Cheetah (Without it Even Noticing). In: *Proceedings of the VLDB Endowment* 3 (2010), Nr. 1-2, S. 515–529
- [DRW06] DUBOC, Leticia ; ROSENBLUM, David S. ; WICKS, Tony: A Framework for Modelling and Analysis of Software Systems Scalability. In: *ACM International Conference on Software Engineering*, 2006, S. 949–952. <http://dx.doi.org/10.1145/1134285.1134460>
- [DSB⁺10] DAS, Sudipto ; SISMANIS, Yannis ; BEYER, Kevin S. ; GERMULLA, Rainer ; HAAS, Peter J. ; MCPHERSON, John: Ricardo: Integrating R and Hadoop. In: *ACM SIGMOD International Conference on Management of Data*, 2010, S. 987–998. <http://dx.doi.org/10.1145/1807167.1807275>

- [DT05] DALAL, Navneet ; TRIGGS, Bill: Histograms of oriented gradients for human detection. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 1, 2005, S. 886–893. <http://dx.doi.org/10.1109/CVPR.2005.177>
- [Eak98] EAKINS, John P.: *Techniques for Image Retrieval (Library and Information Briefings 85)*. British Library and South Bank University, London, 1998
- [EBB04] EAKINS, John P. ; BRIGGS, Pam ; BURFORD, Bryan: Image Retrieval Interfaces: A User Perspective. In: *International Conference on Image and Video Retrieval*, 2004, S. 628–637. http://dx.doi.org/10.1007/978-3-540-27814-6_73
- [Eid04] EIDENBERGER, Horst: Statistical analysis of content-based MPEG-7 descriptors for image retrieval. In: *Multimedia Systems* 10 (2004), Nr. 2, S. 84–97. <http://dx.doi.org/10.1007/s00530-004-0141-8>
- [Ens93] ENSER, Peter G.: Query analysis in a visual information retrieval context. In: *Journal of Document and Text Management* 1 (1993), Nr. 1, S. 25–52
- [EOW10] EPSHTEIN, Boris ; OFEK, Eyal ; WEXLER, Yonatan: Detecting Text in Natural Scenes with Stroke Width Transform. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2010, S. 2963–2970. <http://dx.doi.org/10.1109/CVPR.2010.5540041>
- [ESC07] *Escort 2007 - EBU System of Classification of Radio and Television Programmes*. October 2007
- [EVGW⁺07] EVERINGHAM, M. ; VAN GOOL, L. ; WILLIAMS, C. ; WINN, J. ; ZISSERMAN, A.: *The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results*. <http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2007/workshop/>. Version: 2007
- [EVGW⁺08] EVERINGHAM, Mark ; VAN GOOL, Luc ; WILLIAMS, Chris ; WINN, John ; ZISSERMAN, Andrew: *The PASCAL Visual Object Classes Challenge 2008 (VOC2008)*. <http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2008/workshop/>. Version: 2008
- [EVGW⁺09] EVERINGHAM, Mark ; VAN GOOL, Luc ; WILLIAMS, Chris ; WINN, John ; ZISSERMAN, Andrew: *The PASCAL Visual Object Classes Challenge*

- 2009 (*VOC2009*). <http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2009/workshop/>. Version: 2009
- [EVGW⁺10] EVERINGHAM, Mark ; VAN GOOL, Luc ; WILLIAMS, Chris ; WINN, John ; ZISSERMAN, Andrew: *The PASCAL Visual Object Classes Challenge 2010 (VOC2010)*. <http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2010/workshop/>. Version: 2010
- [EVGW⁺11] EVERINGHAM, Mark ; VAN GOOL, Luc ; WILLIAMS, Chris ; WINN, John ; ZISSERMAN, Andrew: *The PASCAL Visual Object Classes Challenge 2011 (VOC2011)*. <http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2011/workshop/>. Version: 2011
- [EXI02] ; Japan Electronics and Information Technology Industries Association (Veranst.): *Exchangeable image file format for digital still cameras: Exif Version 2.2*. <http://www.exif.org/Exif2-2.PDF>. Version: April 2002
- [EZW⁺05] EVERINGHAM, Mark ; ZISSERMAN, Andrew ; WILLIAMS, Christopher ; VAN GOOL, Luc ; ALLAN, Moray ; BISHOP, Christopher ; CHAPELLE, Olivier ; DALAL, Navneet ; DESELAERS, Thomas ; DORKÓ, Gyuri ; DUFFNER, Stefan ; EICHHORN, Jan ; FARQUHAR, Jason ; FRITZ, Mario ; GARCIA, Christophe ; GRIFFITHS, Tom ; JURIE, Frederic ; KEYSERS, Daniel ; KOSKELA, Markus ; LAAKSONEN, Jorma ; LARLUS, Diane ; LEIBE, Bastian ; MENG, Hongying ; NEY, Hermann ; SCHIELE, Bernt ; SCHMID, Cordelia ; SEEMANN, Edgar ; SHAWE-TAYLOR, John ; STORKEY, Amos ; SZEDMAK, Sandor ; TRIGGS, Bill ; ULUSOY, Ilkay ; VIITANIEMI, Ville ; ZHANG, Jianguo: The 2005 PASCAL Visual Object Classes Challenge. In: *LNCS: Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Textual Entailment* 3944 (2005), S. 117–176. <http://dx.doi.org/10.1007/11736790>
- [EZWVG06] EVERINGHAM, Mark ; ZISSERMAN, Andrew ; WILLIAMS, Chris ; VAN GOOL, Luc: *The PASCAL Visual Object Class Challenge 2006 (VOC2006) Results*. <http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2006/results.pdf>. Version: September 2006
- [FA91] FREEMAN, William T. ; ADELSON, Edward H.: The Design and Use of Steerable Filters. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13 (1991), Nr. 9, S. 891–906. <http://dx.doi.org/>

- 10.1109/34.93808
- [FBY92] *Kapitel Lexical analysis and stoplists.* In: FRAKES, W. B. ; BAEZA-YATES, R.: *Information retrieval: data structures and algorithms.* Prentice Hall Inc., 1992, S. 102–130
- [FCH⁺08] FAN, Rong-En ; CHANG, Kai-Wei ; HSIEH, Cho-Jui ; WANG, Xiang-Rui ; LIN, Chih-Jen: LIBLINEAR: A Library for Large Linear Classification. In: *The Journal of Machine Learning Research* 9 (2008), S. 1871–1874
- [Fel98] FELLBAUM, Christiane (Hrsg.): *WordNet: An Electronic Lexical Database.* MIT Press, 1998
- [FFFP04] FEI-FEI, Li ; FERGUS, Rob ; PERONA, Pietro: Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. In: *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, 2004, S. 178–187. <http://dx.doi.org/10.1109/CVPR.2004.109>
- [FFFP06] FEI-FEI, Li ; FERGUS, Rob ; PERONA, Pietro: One-Shot Learning of Object Categories. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (2006), Nr. 4, S. 594–611. <http://dx.doi.org/10.1109/TPAMI.2006.79>
- [FFJS08] FERRARI, Vittorio ; FEVRIER, Loic ; JURIE, Frédéric ; SCHMID, Cordelia: Groups of Adjacent Contour Segments for Object Detection. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (2008), Nr. 1, S. 36–51. <http://dx.doi.org/10.1109/TPAMI.2007.1144>
- [FFP05] FEI-FEI, Li ; PERONA, Pietro: A Bayesian Hierarchical Model for Learning Natural Scene Categories. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 2, 2005, S. 524–531. <http://dx.doi.org/10.1109/CVPR.2005.16>
- [FGL04] FAN, Jianping ; GAO, Yuli ; LUO, Hangzai: Multi-Level Annotation of Natural Scenes Using Dominant Image Components and Semantic Concepts. In: *ACM International Conference on Multimedia*, 2004, S. 540–547. <http://dx.doi.org/10.1145/1027527.1027660>
- [FGL07] FAN, Jianping ; GAO, Yuli ; LUO, Hangzai: Hierarchical Classification for Automatic Image Annotation. In: *ACM SIGIR International Conference*

- on Research and Development in Information Retrieval, 2007, S. 111–118.
<http://dx.doi.org/10.1145/1277741.1277763>
- [FGLX04] FAN, Jianping ; GAO, Yuli ; LUO, Hangzai ; XU, Guangyou: Automatic Image Annotation by Using Concept-Sensitive Salient Objects for Image Content Representation. In: *ACM SIGIR International Conference on Research and Development in Information Retrieval*, 2004, S. 361–368. <http://dx.doi.org/10.1145/1008992.1009055>
- [Fid97] FIDEL, Raya: The image retrieval task: implications for the design and evaluation of image databases. In: *The New Review of Hypermedia and Multimedia* 3 (1997), S. 181–199
- [FK09] FERZLI, R. ; KARAM, L. J.: A No-Reference Objective Image Sharpness Metric Based on the Notion of Just Noticeable Blur (JNB). In: *IEEE Transactions on Image Processing* 18 (2009), Nr. 4, S. 717–728. <http://dx.doi.org/10.1109/TIP.2008.2011760>
- [FMH06] FLUHR, Christian ; MOËLLIC, Pierre-Alain ; HEDE, Patrick: Usage-oriented Multimedia Information Retrieval Technological Evaluation. In: *ACM International Workshop on Multimedia Information Retrieval*, 2006, S. 301–306. <http://dx.doi.org/10.1145/1178677.1178719>
- [FML04] FENG, S. L. ; MANMATHA, R. ; LAVRENKO, V.: Multiple Bernoulli relevance models for image and video annotation. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 2, 2004, S. 1002–1009. <http://dx.doi.org/10.1109/CVPR.2004.1315274>
- [FP12] FORSYTH, David A. ; PONCE, Jean: *Computer Vision - A Modern Approach*. Pearson Education Limited, 2012
- [FSMST05] FARQUHAR, J.D.R. ; SZEDMAK, Sandor ; MENG, Hongying ; SHAWETAYLOR, John: Improving "bag-of-keypoints"image categorisation: Generative Models and PDF-Kernels / Department of Electronics and Computer Science, University of Southampton. 2005. – Forschungsbericht
- [FSN⁺95] FLICKNER, Myron ; SAWHNEY, Harpreet ; NIBLACK, Wayne ; ASHLEY, Jonathan ; HUANG, Qian ; DOM, Byron ; GORKANI, Monika ; HAFNER, Jim ; LEE, Denis ; PETKOVIC, Dragutin ; STEELE, David ; YANKER, Peter: Query by image and video content: the QBIC system. In: *IEEE Computer* 28 (1995), Nr. 9, S. 23–32. <http://dx.doi.org/10.1109/2.410146>

- [FYS⁺09] FAN, Jianping ; YANG, Chunlei ; SHEN, Yi ; BABAGUCHI, Noboru ; LUO, Hangzai: Leveraging large-scale weakly-tagged images to train inter-related classifiers for multi-label annotation. In: *ACM Workshop on Large-scale Multimedia Retrieval and Mining*, 2009, S. 27–34. <http://dx.doi.org/10.1145/1631058.1631066>
- [Gal09] GALEA, Andrei: *Entwurf und Implementierung eines Frameworks zur Extraktion von Features aus Bildern*, Friedrich-Alexander-Universität Erlangen-Nürnberg, Diplomarbeit, 2009
- [GD05] GRAUMAN, Kristen ; DARRELL, Trevor: Pyramid Match Kernels: Discriminative Classification with Sets of Image Features / MIT CSAIL, Cambridge, MA, USA. 2005. – Forschungsbericht. AIM-2005-007
- [GD07] GRAUMAN, Kristen ; DARRELL, Trevor: The Pyramid Match Kernel: Efficient Learning with Sets of Features. In: *Journal of Machine Learning Research* 8 (2007), S. 725–760
- [GEW06] GUERTS, Pierre ; ERNST, Damien ; WEHENKEL, Louis: Extremely randomized trees. In: *Machine Learning* 63 (2006), Nr. 1, S. 3–42. <http://dx.doi.org/10.1007/s10994-006-6226-1>
- [GG09] GU, Yunhong ; GROSSMANN, Robert: Sector and Sphere: The Design and Implementation of a High Performance Data Cloud. In: *Theme Issue of the Philosophical Transactions of the Royal Society A: Crossing Boundaries: Computational Science, E-Science and Global E-Infrastructure* 367 (2009), Nr. 1897, S. 2429–2445. <http://dx.doi.org/10.1098/rsta.2009.0053>
- [GGL03] GHEMAWAT, Sanjay ; GOBIOFF, Howard ; LEUNG, Shun-Tak: The Google File System. In: *ACM Symposium on Operating System Principles*, 2003, S. 29–43. <http://dx.doi.org/10.1145/945445.945450>
- [GGVS08] GEMERT, Jan C. ; GEUSEBROEK, Jan-Mark ; VEENMAN, Cor J. ; SMEULDERS, Arnold W. M.: Kernel Codebooks for Scene Categorization. In: *European Conference on Computer Vision*, 2008, S. 696–709. http://dx.doi.org/10.1007/978-3-540-88690-7_52
- [GHP07] GRIFFIN, Greg ; HOLUB, Alex ; PERONA, Pietro: Caltech-256 Object Category Dataset / California Institute of Technology. Version: March 2007. <http://resolver.caltech.edu/CaltechAUTHORS:CNS-TR-2007-001> (7694). – Forschungsbericht. – Online-Ressource

- [GIM99] GIONIS, Aristides ; INDYK, Piotr ; MOTWANI, Rajeev: Similarity Search in High Dimensions via Hashing. In: *International Conference on Very Large Databases*, 1999, S. 518–529
- [GJA10] GUDMUNDSSON, Gylfi ; JÓNSSON, Björn ; AMSALEG, Laurent: A Large-Scale Performance Study of Cluster-Based High-Dimensional Indexing. In: *ACM International Workshop on Very-Large-Scale Multimedia Corpus, Mining and Retrieval*, 2010, S. 31–36. <http://dx.doi.org/10.1145/1878137.1878145>
- [GL97] GLOVER, Fred ; LAGUNA, Manuel: Tabu Search. In: *Journal of Computational Biology* 16 (1997), Nr. 12, S. 1689–1703. <http://dx.doi.org/10.1089/cmb.2007.0211>
- [GLDS96] GROPP, William ; LUSK, Ewing ; DOSS, Nathan ; SKJELLUM, Anthony: A high-performance, portable implementation of the MPI message passing interface standard. In: *Parallel Computing* 22 (1996), Nr. 6, S. 789–828. [http://dx.doi.org/10.1016/0167-8191\(96\)00024-5](http://dx.doi.org/10.1016/0167-8191(96)00024-5)
- [GMS08] GAIDON, Adrien ; MARSZAŁEK, Marcin ; SCHMID, Cordelia: *The PASCAL Visual Object Classes Challenge 2008 submission*. <http://lear.inrialpes.fr/people/gaidon/gaidon.pdf>. Version: September 2008
- [GMU96] GOOL, Luc van ; MOONS, Theo ; UNGUREANU, Dorin: Affine / photometric invariants for planar intensity patterns. In: *European Conference on Computer Vision*, 1996, S. 642–651. <http://dx.doi.org/10.1007/BFb0015518>
- [GN09] GEHLER, Peter ; NOWOZIN, Sebastian: On Feature Combination for Multiclass Object Classification. In: *IEEE International Conference on Computer Vision*, 2009, S. 221–228. <http://dx.doi.org/10.1109/ICCV.2009.5459169>
- [GO02] GREISDORF, Howard ; O'CONNOR, Brian: Modelling what users see when they look at images: a cognitive viewpoint. In: *Journal of Documentation* 58 (2002), Nr. 1, S. 6–29. <http://dx.doi.org/10.1108/00220410210425386>
- [GP08] GRIFFIN, Gregory ; PERONA, Pietro: Learning and Using Taxonomies for Fast Visual Categorization. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2008, S. 1–8. <http://dx.doi.org/10.1109/CVPR.2008.4587551>

- org/10.1109/CVPR.2008.4587410
- [GS01] GOODRUM, Abby ; SPINK, Amanda: Image searching on the Excite Web search engine. In: *Information Processing & Management* 37 (2001), Nr. 2, S. 295–311. [http://dx.doi.org/10.1016/S0306-4573\(00\)00033-9](http://dx.doi.org/10.1016/S0306-4573(00)00033-9)
- [GSK98] GOMES, Carla P. ; SELMAN, Bart ; KAUTZ, Henry: Boosting Combinatorial Search Through Randomization. In: *AAAI International Joint Conference on Artificial Intelligence*, 1998, S. 431–437
- [Gus94] GUSTAVSON, David B.: The Many Dimensions of Scalability. In: *COMPCON*, 1994, S. 60–63. <http://dx.doi.org/10.1109/CMPCON.1994.282944>
- [Ham50] HAMMING, Richard W.: Error Detecting and Error Correcting Codes. In: *The Bell System Technical Journal* 29 (1950), Nr. 2, S. 147–160
- [Han07] HANBURY, Allan: A Study of Vocabularies for Image Annotation. In: *International Conference on Semantic and Digital Media Technologies*, 2007, S. 284–287. <http://dx.doi.org/10.1007/978-3-540-77051-0>
- [HE08a] HAYS, James ; EFROS, Alexei A.: IM2GPS: estimating geographic information from a single image. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2008, S. 1–8. <http://dx.doi.org/10.1109/CVPR.2008.4587784>
- [HE08b] HAYS, James ; EFROS, Alexei A.: Scene Completion Using Millions of Photographs. In: *Communications of the ACM* 51 (2008), Nr. 10, S. 87–94. <http://dx.doi.org/10.1145/1400181.1400202>
- [Hen05] HENDERSON, John M.: Introduction to Real-World Scene Perception. In: *Visual Cognition: Special Issue on Real-World Scene Perception* 12 (2005), S. 849–851
- [Hen08] HENRICH, Andreas: *Information Retrieval 1 - Grundlagen, Modelle und Anwendungen*. Otto-Friedrich-Universität Bamberg, Lehrstuhl für Medieninformatik
- [Hil90] HILL, Mark D.: What is Scalability? In: *SIGARCH Computer Architecture News* 18 (1990), Nr. 4, S. 18–21. <http://dx.doi.org/10.1145/121973.121975>

- [Hir75] HIRSCHBERG, Daniel S.: A Linear Space Algorithm for Computing Maximal Common Subsequences. In: *Communications of the ACM* 18 (1975), Nr. 6, S. 341–343. <http://dx.doi.org/10.1145/360825.360861>
- [HL08] HUISKES, Mark J. ; LEW, Michael S.: The MIR Flickr Retrieval Evaluation. In: *ACM International Conference on Multimedia Information Retrieval*, 2008, S. 39–43. <http://dx.doi.org/10.1145/1460096.1460104>
- [HL09] HOFFMANN, Simon ; LIENHART, Rainer ; HOFFMANN, Simon (Hrsg.) ; LIENHART, Rainer (Hrsg.): *OpenMP: Eine Einführung in die parallele Programmierung mit C/C++*. Springer, 2009. <http://dx.doi.org/10.1007/978-3-540-73123-8>
- [HNP95] HELLERSTEIN, Joseph M. ; NAUGHTON, Jeffrey F. ; PFEFFER, Avi: Generalized Search Trees for Database Systems. In: *International Conference on Very Large Data Bases*, 1995, S. 562–573
- [Hof09] HOFFMANN, Gernot: *CIELab Color Space*. <http://www.fho-emden.de/~hoffmann/cielab03022003.pdf>. Version: September 2009
- [Hol07] HOLUB, Alex: *Discriminative vs. Generative Object Recognition: Objects, Faces and the Web*, California Institute of Technology, Dissertation, April 2007
- [HSWW03] HOLLINK, Laura ; SCHREIBER, Guus ; WIELEMAKER, Jan ; WIELINGA, Bob: Semantic Annotation of Image Collections. In: *Knowledge Capture 2003, Knowledge Markup and Semantic Annotation Workshop*, 2003
- [HSWW04] HOLLINK, Laura ; SCHREIBER, A. T. ; WIELINGA, Bob J. ; WORRING, M.: Classification of user image descriptions. In: *International Journal of Human-Computer Studies* 61 (2004), Nr. 5, S. 601–626. <http://dx.doi.org/10.1016/j.ijhcs.2004.03.002>
- [HTL10] HUISKES, Mark J. ; THOME, Bart ; LEW, Michael S.: New Trends and Ideas in Visual Concept Detection. In: *ACM International Conference on Multimedia Information Retrieval*, 2010, S. 527–536. <http://dx.doi.org/10.1145/1743384.1743475>
- [HTV11] HOLLINK, Vera ; TSIKRIKA, Theodora ; VRIES, Arjen P.: Semantic Search Log Analysis: A Method and a Study on Professional Image Search. In: *Journal of the American Society for Information Science and Technology*

- 62 (2011), Nr. 4, S. 691–713. <http://dx.doi.org/10.1002/asi.21484>
- [IAS] *Image Annotation on the Semantic Web.* <http://www.w3.org/2005/Incubator/mmsem/XGR-image-annotation/>
- [IPT99] ; International Press Telecommunications Council (Veranst.): *IPTC-NAA Information Interchange Model Version 4.* <http://www.iptc.org/std/IIM/4.1/specification/IIMV4.1.pdf>. Version: July 1999
- [IPT08] ; International Press Telecommunications Council (Veranst.): *IPTC Standard Photo Metadata 2008, IPTC Core Specification Version 1.1, IPTC Extension Specification Version 1.0.* http://iptc.cms.apa.at/std/photometadata/2008/specification/IPTC-PhotoMetadata-2008_2.pdf. Version: July 2008
- [Jar89] JARO, Matthew A.: Advances in record linkage methodology as applied to the 1985 census of Tampa Florida. In: *Journal of the American Statistical Association* 84 (1989), Nr. 406, S. 414–420
- [JC97] JIANG, Jay J. ; CONRATH, David W.: Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In: *International Conference Research on Computational Linguistics*, 1997
- [JC02] *Kapitel Concepts and Techniques for Indexing Visual Semantics.* In: JAIMES, Alejandro ; CHANG, Shih-Fu: *Image Databases: Search and Retrieval of Digital Imagery.* John Wiley & Sons, 2002, S. 497–565. <http://dx.doi.org/10.1002/0471224634.ch17>
- [JCS04] JIN, Rong ; CHAI, Joyce Y. ; SI, Luo: Effective Automatic Image Annotation Via A Coherent Language Model and Active Learning. In: *ACM International Conference on Multimedia*, 2004, S. 892–899. <http://dx.doi.org/10.1145/1027527.1027732>
- [JDS08] JÉGOU, Hervé ; DOUZE, Matthijs ; SCHMID, Cordelia: Hamming Embedding and Weak Geometric Consistency for Large Scale Image Search. In: *European Conference on Computer Vision*, 2008, S. 304–317. http://dx.doi.org/10.1007/978-3-540-88682-2_24
- [JDS09] JÉGOU, Hervé ; DOUZE, Matthijs ; SCHMID, Cordelia: Packing Bag-of-features. In: *IEEE International Conference on Computer Vision*, 2009, S. 2357–2364. <http://dx.doi.org/10.1109/ICCV.2009.5459419>

- [JH97] JOHNSON, Andrew E. ; HERBERT, Martial: Recognizing Objects by Matching Oriented Points. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 1997, S. 684–689. <http://dx.doi.org/10.1109/CVPR.1997.609400>
- [JH98] JAAKKOLA, Tommi S. ; HAUSSLER, David: Exploiting generative models in discriminative classifiers. In: *Advances in Neural Information Processing Systems*, 1998, S. 487–493
- [JJ93] JORDAN, Michael I. ; JACOBS, Robert A.: Hierarchical mixtures of experts and the EM algorithm. In: *Proceedings of International Joint Conference on Neural Networks* Bd. 2, 1993, S. 1339–1344. <http://dx.doi.org/10.1109/IJCNN.1993.716791>
- [JJ05] JÖRGENSEN, Corinne ; JÖRGENSEN, Peter: Image Querying by Image Professionals. In: *Journal of the American Society for Information Science and Technology* 56 (2005), Nr. 12, S. 1346–1359. <http://dx.doi.org/10.1002/asi.v56:12>
- [JKWA05] JIN, Yohan ; KHAN, Latifur ; WANG, Lei ; AWAD, Mamoun: Image Annotations By Combining Multiple Evidence & WordNet. In: *ACM International Conference on Multimedia*, 2005, S. 706–715. <http://dx.doi.org/10.1145/1101149.1101305>
- [JLM03] JEON, Jiwoon ; LAVRENKO, V. ; MANMATHA, R.: Automatic Image Annotation and Retrieval using Cross-Media Relevance Models. In: *ACM SIGIR International Conference on Research and Development in Information Retrieval*, 2003, S. 119–126. <http://dx.doi.org/10.1145/860435.860459>
- [JM04] JEON, Jiwoon ; MANMATHA, R.: Using Maximum Entropy for Automatic Image Annotation. In: *ACM International Conference on Image and Video Retrieval*, 2004, S. 24–32. http://dx.doi.org/10.1007/11562214_4
- [Jör96] JÖRGENSEN, Corinne: Indexing Images: Testing an Image Description Template. In: *ASIS 1996 Annual Conference Proceedings*, 1996, S. 209–213
- [Jör98] JÖRGENSEN, Corinne: Attributes of images in describing tasks. In: *Information Processing & Management* 34 (1998), Nr. 2-3, S. 161–174. [http://dx.doi.org/10.1016/S0306-4573\(97\)00077-0](http://dx.doi.org/10.1016/S0306-4573(97)00077-0)

- [JS05] JANSEN, Bernard J. ; SPINK, Amanda: An analysis of web searching by European AlltheWeb.com users. In: *Information Processing & Management* 41 (2005), Nr. 2, S. 361–381. [http://dx.doi.org/10.1016/S0306-4573\(03\)00067-0](http://dx.doi.org/10.1016/S0306-4573(03)00067-0)
- [JSP03] JANSEN, Bernard J. ; SPINK, Amanda ; PEDERSEN, Jan: An Analysis of Multimedia Searching on AltaVista. In: *ACM SIGMM International Workshop on Multimedia Information Retrieval*, 2003, S. 186–192. <http://dx.doi.org/10.1145/973264.973294>
- [JSS00] JANSEN, Bernard J. ; SPINK, Amanda ; SARACEVIS, Tefko: Real life, real users, and real needs: a study and analysis of user queries on the web. In: *Information Processing & Management* 36 (2000), Nr. 2, S. 207–227. [http://dx.doi.org/10.1016/S0306-4573\(99\)00056-4](http://dx.doi.org/10.1016/S0306-4573(99)00056-4)
- [JSW11] JÖRGENSEN, Corinne ; STVILIA, Besiki ; WU, Shuheng: Assessing the Quality of Socially Created Metadata to Image Indexing. In: *Proceedings of the American Society for Information Science and Technology* Bd. 48, 2011, S. 1–4. <http://dx.doi.org/10.1002/meet.2011.14504801325>
- [Kah01] KAHLBRANDT, Bernd: *Software-Engineering mit der Unified Modeling Language*. 2. Springer, 2001
- [KB96] KHOSHAFIAN, Setrag ; BAKER, A. B.: *MultiMedia and Imaging Databases*. Morgan Kaufmann Publishers, Inc., 1996
- [KD87] KOENDERINK, J. J. ; DOORN, A. J.: Representation of Local Geometry in the Visual System. In: *Biological Cybernetics* 55 (1987), Nr. 6, S. 367–375. <http://dx.doi.org/10.1007/BF00318371>
- [Ken06] KENNEDY, Lyndon: LSCOM Lexicon Definitions and Annotations Version 1.0, DTO Challenge Workshop on Large Scale Concept Ontology for Multimedia / Columbia University. 2006. – Forschungsbericht. 217-2006-3
- [KG09] KULIS, Brian ; GRAUMAN, Kristen: Kernelized Locality-Sensitive Hashing for Scalable Image Search. In: *IEEE International Conference on Computer Vision*, 2009, S. 2130–2137. <http://dx.doi.org/10.1109/ICCV.2009.5459466>

- [KGV83] KIRKPATRICK, Scott ; GELATT, C. D. ; VECCHI, Mario P.: Optimization by Simulated Annealing. In: *Science* 220 (1983), Nr. 4598, S. 671–680. <http://dx.doi.org/10.1126/science.220.4598.671>
- [KHD98] KITTLER, Josef ; HATEF, Mohamad ; DUIN, Robert P. W. ; MATAS, Jiri: On Combining Classifiers. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20 (1998), Nr. 3, S. 226–239. <http://dx.doi.org/10.1109/34.667881>
- [KLB01] KANUNGO, Tapas ; LEE, Chang H. ; BRADFORD, Roger: What fraction of images on the web contain text? In: *International Workshop on Web Document Analysis*, 2001, S. 43–46
- [Kos04] KOSCH, Harald: *Distributed Multimedia Database Technologies Supported by MPEG-7 and MPEG-21*. CRC Press, 2004
- [Kos10] KOSCH, Harald: Optimizing Similarity-based Image Joins in a Multimedia Database. In: *ACM International Workshop on Very-Large-Scale Multimedia Corpus, Mining and Retrieval*, 2010, S. 37–42. <http://dx.doi.org/10.1145/1878137.1878146>
- [KPD96] KENNEL, Andrea ; PERROCHON, Louis ; DARVISHI, Alireza: WAB: World Wide Web access for blind and visually impaired computer users. In: *ACM SIGCAPH Computers and the Physically Handicapped*, 1996, S. 10–15. <http://dx.doi.org/10.1145/231674.231675>
- [Kra88] KRAUSE, M.G.: Intellectual Problems of Indexing Picture Collections. In: *Audiovisual Librarian* 14 (1988), Nr. 2, S. 73–81
- [Kru83] Kapitel An Overview of Sequence Comparison. In: KRUSKAL, Joseph B.: *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison-Wesley, 1983, S. 1–44
- [KS97] KATAYAMA, Norioa ; SATOH, Shin’ichi: The SR-tree: An Index Structure for High-Dimensional Nearest Neighbor Queries. In: *ACM SIGMOD International Conference on Management of Data*, 1997, S. 369–380. <http://dx.doi.org/10.1145/253260.253347>
- [KS04] KE, Yan ; SUKTHANKAR, Rahul: PCA-SIFT: A More Distinctive Representation for Local Image Descriptors. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 2, 2004, S. 506–513.

- <http://dx.doi.org/10.1109/CVPR.2004.1315206>
- [KS07] KUMAR, Abnkita ; SMINCHISESCU, Cristian: Support Kernel Machines for Object Recognition. In: *IEEE International Conference on Computer Vision*, 2007, S. 1–8. <http://dx.doi.org/10.1109/ICCV.2007.4409065>
- [LC98] Kapitel Combining Local Context and WordNet Similarity for Word Sense Identification. In: LEACOCK, Claudia ; CHODOROW, Martin: *Wordnet: An Electronic Lexical Database*. MIT Press, 1998, S. 265–284
- [LCB⁺04] LANCKRIET, Gert R. G. ; CRISTIANINI, Nello ; BARTLETT, Peter ; GHAOUI, Laurent E. ; JORDAN, Michael I.: Learning the Kernel Matrix with Semidefinite Programming. In: *Journal of Machine Learning Research* 5 (2004), S. 27–72
- [LDL05] LIANG, Jian ; DOERMANN, David ; LI, Huiping: Camera-based analysis of text and documents: a survey. In: *International Journal on Document Analysis and Recognition* 7 (2005), S. 84–104. – 10.1007/s10032-004-0138-z. <http://dx.doi.org/10.1007/s10032-004-0138-z>. – ISSN 1433–2833
- [Lev66] LEVENSHTEIN, Vladimir I.: Binary codes capable of correcting deletions, insertions, and reversals. In: *Soviet Physics Doklady* 10 (1966), Nr. 8, S. 707–710
- [Lin68] LINDSAY, Kenneth C.: Computer Input Form for Art Works: Problems and Possibilities. In: *Computers and Their Potential Applications in Museums*, 1968, S. 19–35
- [Lin98] LIN, Dekang: An Information-Theoretic Definition of Similarity. In: *International Conference on Machine Learning*, 1998, S. 296–304
- [LLS04] LEIBE, Bastian ; LEONARDIS, Ales ; SCHIELE, Bernt: Combined object categorization and segmentation with an implicit shape model. In: *Workshop on Statistical Learning in Computer Vision*, 2004, S. 17–32
- [LMJ03] LAVRENKO, V. ; MANMATHA, R. ; JEON, Jiwoon: A model for learning the semantics of pictures. In: *Advances in Neural Information Processing Systems*, 2003
- [Loc90] LOCKE, John: *An Essay Concerning the Humane Understanding*. T. Basset, E. Mory (Gemeinfrei), 1690

- [Low04] LOWE, David G.: Distinctive Image Features from Scale-Invariant Key-points. In: *International Journal of Computer Vision* 60 (2004), Nr. 2, S. 91–110. <http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94>
- [LPS⁺03] LUCAS, S.M. ; PANARETOS, A. ; SOSA, L. ; TANG, A. ; WONG, S. ; YOUNG, R.: ICDAR 2003 Robust Reading Competitions. In: *International Conference on Document Analysis and Recognition*, 2003, S. 682–687. <http://dx.doi.org/10.1109/ICDAR.2003.1227749>
- [LPS⁺05] LUCAS, Simon M. ; PANARETOS, Alex ; SOSA, Luis ; TANG, Anthony ; WONG, Shirley ; YOUNG, Robert ; ASHIDA, Kazuki ; NAGAI, Hiroki ; OKAMOTO, Masayuki ; YAMAMOTO, Hiroaki ; MIYAO, Hidetoshi M. ; ZHU, JunMin ; OU, WuWen ; WOLF, Christian ; JOLION, Jean-Michel ; TODORAN, Leon ; WORRING, Marcel ; LIN, Xiaofan: ICDAR 2003 Robust Reading Competitions: Entries, Results, and Future Directions. In: *International Journal on Document Analysis and Recognition* 7 (2005), Nr. 2-3, S. 105–122. <http://dx.doi.org/10.1007/s10032-004-0134-3>
- [LS04] LIU, Hugo ; SINGH, Push: ConceptNet - A Practical Commonsense Reasoning Tool-Kit. In: *BT Technology Journal* 22 (2004), Nr. 4, S. 211–226. <http://dx.doi.org/10.1023/B:BTTJ.0000047600.45421.6d>
- [LSP05] LAZEBNIK, Svetlana ; SCHMID, Cordelia ; PONCE, Jean: A Sparse Texture Representation Using Affine-Invariant Regions. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (2005), Nr. 8, S. 1265–1278. <http://dx.doi.org/10.1109/TPAMI.2005.151>
- [LSP06] LAZEBNIK, Svetlana ; SCHMID, Cordelia ; PONCE, Jean: Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 2, 2006, S. 2169–2178. <http://dx.doi.org/10.1109/CVPR.2006.68>
- [Luk93] LUKE, Edward A.: Defining and Measuring Scalability. In: *Scalable Parallel Libraries Conference*, 1993, S. 183–186. <http://dx.doi.org/10.1109/SPLC.1993.365568>
- [LW06] LI, Jia ; WANG, James Z.: Real-Time Computerized Annotation of Pictures. In: *ACM International Conference on Multimedia*, 2006, S. 911–920. <http://dx.doi.org/10.1145/1180639.1180841>

- [LW08] LI, Jia ; WANG, James Z.: Real-Time Computerized Annotation of Pictures. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (2008), Nr. 6, S. 985–1002. <http://dx.doi.org/10.1109/TPAMI.2007.70847>
- [LWZ⁺08] LI, Xiaowei ; WU, Changchang ; ZACH, Christopher ; LAZEBNIK, Svetlana ; FRAHM, Jan-Michael: Modeling and Recognition of Landmark Image Collections Using Iconic Scene Graphs. In: *European Conference on Computer Vision* Bd. 5302/2008, 2008, S. 427–440. http://dx.doi.org/10.1007/978-3-540-88682-2_33
- [LYCR05] LIU, Song ; YI, Haoran ; CHIA, Liang-Tien ; RAJAN, Deepu: Adaptive Hierarchical Multi-class SVM Classifier for Texture-based Image Classification. In: *IEEE International Conference on Multimedia and Expo*, 2005, S. 4. <http://dx.doi.org/10.1109/ICME.2005.1521640>
- [LZ00] LOPRESTI, Daniel ; ZHOU, Jiangying: Locating and Recognizing Text in WWW Images. In: *Information Retrieval* 2 (2000), Nr. 2-3, S. 177–206. <http://dx.doi.org/10.1023/A:1009954710479>
- [Mar88] MARKEY, Karen: Access to Iconographical Research Collections. In: *Library Trends: Linking Art Objects and Art Information* 2 (1988), Nr. 37, S. 154–174
- [MC12] MANDUCHI, Roberto ; COUGHLAN, James: (Computer) Vision Without Sight. In: *Communications of the ACM* 55 (2012), Nr. 1, S. 96–104. <http://dx.doi.org/10.1145/2063176.2063200>
- [McC] McCUALEY, Trevor: *Understanding the Transformation Matrix in Flash* 8. <http://www.senocular.com/flash/tutorials/transformmatrix/> – Online-Ressource, Abruf: 04.03.2011
- [MCMP02] MATAS, J. ; CHUM, O. ; MARTIN, U. ; PAJDLA, T.: Robust wide baseline stereo from maximally stable extremal regions. In: *British Machine Vision Conference*, 2002, S. 384–393. <http://dx.doi.org/10.1016/j.imavis.2004.02.006>
- [Mer09] MERCER, James: Functions of Positive and Negative Type, and their Connection with the Theory of Integral Equations. In: *Philosophical Transactions of the Royal Society* 209 (1909), S. 415–446. <http://dx.doi.org/10.1098/rsta.1909.0016>

- [Met78] METZ, Charles E.: Basic Principles of ROC Analysis. In: *Seminars in Nuclear Medicine* 8 (1978), Nr. 4, S. 283–298. [http://dx.doi.org/10.1016/S0001-2998\(78\)80014-2](http://dx.doi.org/10.1016/S0001-2998(78)80014-2)
- [Mey07] MEYERHÖFER, Marcus B.: *Messung und Verwaltung von Softwarekomponenten für die Performancevorhersage*, Friedrich-Alexander-Universität Erlangen-Nürnberg, Dissertation, Juli 2007
- [MGPW05] MARÉE, Raphael ; GUERTS, Pierre ; PIATER, Justus ; WEHENKEL, Louis: Random Subwindows for Robust Image Classification. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 1, 2005, S. 34–40. <http://dx.doi.org/10.1109/CVPR.2005.287>
- [MHG10] McCANDLESS, Michael ; HATCHER, Erik ; GOSPODNETIĆ, Otis: *Lucene in Action*. 2. Manning, 2010
- [Mil95] MILLER, George A.: WordNet: A Lexical Database for English. In: *Communications of the ACM* 38 (1995), Nr. 11, S. 39–41. <http://dx.doi.org/10.1145/219717.219748>
- [MM04] METZLER, Donald ; MANMATHA, R.: An Inference Network Approach to Image Retrieval. In: *ACM International Conference on Image and Video Retrieval*, 2004, S. 42–50. <http://dx.doi.org/10.1007/b98923>
- [MMMP02] MÜLLER, Henning ; MARCHAND-MAILLET, Stéphane ; PUN, Thierry: The Truth about Corel - Evaluation in Image Retrieval. In: *ACM International Conference on Image and Video Retrieval*, 2002, S. 38–49. <http://dx.doi.org/10.1007/3-540-45479-9>
- [MMR⁺01] MÜLLER, Klaus-Robert ; MIKA, Sebastian ; RÄTSCH, Gunnar ; TSUDA, Koji ; SCHÖLKOPF, Bernhard: An Introduction to Kernel-Based Learning Algorithms. In: *IEEE Transactions on Neural Networks* 12 (2001), Nr. 2, S. 181–201. <http://dx.doi.org/10.1109/72.914517>
- [MMS⁺01] MÜLLER, Henning ; MÜLLER, Wolfgang ; SQUIRE, David M. ; MARCHAND-MAILLET, Stéphane ; PUN, Thierry: Performance evaluation in content-based image retrieval: overview and proposals. In: *Pattern Recognition Letters* 22 (2001), Nr. 5, S. 593–601. [http://dx.doi.org/10.1016/S0167-8655\(00\)00118-5](http://dx.doi.org/10.1016/S0167-8655(00)00118-5)

- [Mou10] MOUNT, David M.: *ANN Programming Manual*. College Park, Maryland: Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, 2010
- [MR05] MAGALHAES, Joao ; RÜGER, Stefan: Mining Multimedia Salient Concepts for Incremental Information Extraction. In: *ACM SIGIR International Conference on Research and Development in Information Retrieval*, 2005, S. 641–642. <http://dx.doi.org/10.1145/1076034.1076168>
- [MRR⁺53] METROPOLIS, Nicholas ; ROSENBLUTH, Arianna W. ; ROSENBLUTH, Marshall N. ; TELLER, Augusta H. ; TELLER, Edward: Equation of State Calculations by Fast Computing Machines. In: *The Journal of Chemical Physics* 21 (1953), Nr. 6, S. 1087–1092. <http://dx.doi.org/10.1063/1.1699114>
- [Mös94] MÖSSENBÖCK, Hanspeter: Extensibility in the Oberon System. In: *Nordic Journal of Computing* 1 (1994), Nr. 1, S. 77–93
- [MS04] MIKOLAJCZYK, Krystian ; SCHMID, Cordelia: Scale & Affine Invariant Interest Point Detectors. In: *International Journal of Computer Vision* 60 (2004), Nr. 1, S. 63–86. <http://dx.doi.org/10.1023/B:VISI.0000027790.02288.f2>
- [MS05] MIKOLAJCZYK, Krystian ; SCHMID, Cordelia: A Performance Evaluation of Local Descriptors. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (2005), Nr. 10, S. 1615–1630. <http://dx.doi.org/10.1109/TPAMI.2005.188>
- [MS07] MARSZAŁEK, Marcin ; SCHMID, Cordelia: Semantic Hierarchies for Visual Object Recognition. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2007, S. 1–7. <http://dx.doi.org/10.1109/CVPR.2007.383272>
- [MS08] MARSZAŁEK, Marcin ; SCHMID, Cordelia: Constructing Category Hierarchies for Visual Recognition. In: *European Conference on Computer Vision* Bd. IV, Springer, October 2008, S. 479–491. http://dx.doi.org/10.1007/978-3-540-88693-8_35
- [MSC⁺06] MUSÉ, Pablo ; SUR, Frédéric ; CAO, Frédéric ; GOUSSEAU, Yann ; MOREL, Jean-Michel: An A Contrario Decision Method for Shape Element Recognition. In: *International Journal of Computer Vision* 69 (2006), Nr.

- 3, S. 295–315. <http://dx.doi.org/10.1007/s11263-006-7546-0>
- [MSHW07] MARSZAŁEK, Marcin ; SCHMID, Cordelia ; HARZALLAH, Hedi ; WEIJER, Joost van d.: Learning Object Representations for Visual Object Class Recognition. In: *Pascal Visual Recognition Challenge Workshop in Conjunction with ICCV*, 2007
- [MTEF06] MURPHY, Kevin ; TORRALBA, Antonio ; EATON, Daniel ; FREEMAN, William: Object detection and localization using local and global features. In: *LNCS: Toward Category-Level Object Recognition* 4170 (2006), S. 382–400. http://dx.doi.org/10.1007/11957959_20
- [MTO99] MORI, Y. ; TAKAHASHI, H. ; OKA, R.: Image-to-word transformation based on dividing and vector quantizing images with words. In: *International Workshop on Multimedia Intelligent Storage and Retrieval Management*, 1999
- [MTS⁺05] MIKOLAJCZYK, Krystian ; TUYTELAARS, Tinne ; SCHMID, Cordelia ; ZISSERMAN, Andrew ; MATAS, J. ; SCHAFFALITZKY, Frederik ; KADIR, Thomas ; VAN GOOL, Luc: A Comparison of Affine Region Detectors. In: *International Journal of Computer Vision* 65 (2005), Nr. 1-2, S. 43–72. <http://dx.doi.org/10.1007/s11263-005-3848-x>
- [MW03] MEYER-WEGENER, Klaus: *Multimediale Datenbanken*. Bd. 2. B. G. Teubner, 2003. – ISBN 3-519-12419-X
- [MY09] MOREL, Jean-Michel ; YU, Guoshen: ASIFT: A New Framework for Fully Affine Invariant Image Comparison. In: *SIAM Journal on Imaging Sciences* 2 (2009), Nr. 2, S. 438–469. <http://dx.doi.org/10.1137/080732730>
- [Nav01] NAVARRO, Gonzalo: A guided tour to approximate string matching. In: *ACM Computing Surveys* 33 (2001), Nr. 1, S. 31–88. <http://dx.doi.org/10.1145/375360.375365>
- [NDMW11a] NAGY, Robert ; DICKER, Anders ; MEYER-WEGENER, Klaus: Definition and Evaluation of the NEOCR Dataset for Natural-Image Text Recognition / University of Erlangen, Dept. of Computer Science. Version: September 2011. <http://www.opus.ub.uni-erlangen.de/opus/volltexte/2011/2859/>. – Forschungsbericht. – Online-Ressource

- [NDMW11b] NAGY, Robert ; DICKER, Anders ; MEYER-WEGENER, Klaus: NEOCR: A Configurable Dataset for Natural Image Text Recognition. In: *Fourth International Workshop on Camera-Based Document Analysis and Recognition at ICDAR 2011*, 2011, S. 53–58
- [NDMW12] *Kapitel NEOCR: A Configurable Dataset for Natural Image Text Recognition.* In: NAGY, Robert ; DICKER, Anders ; MEYER-WEGENER, Klaus: *Camera-Based Document Analysis and Recognition, Lecture Notes in Computer Science*. Bd. 7139. Springer, 2012, S. 150–163. http://dx.doi.org/10.1007/978-3-642-29364-1_12
- [Nie83] NIEMANN, Heinrich: *Klassifikation von Mustern*. Springer Verlag, 1983
- [Nie03] NIEMANN, Heinrich: *Klassifikation von Mustern*. 2. Online
- [NIS06] ANSI/NISO Z39.87 Data Dictionary - Technical Metadata for Digital Still Images. December 2006
- [NL09] NOWAK, Stefanie ; LUKASHEVICH, Hanna: Multilabel Classification Evaluation using Ontology Information. In: *ESWC Workshop on Inductive Reasoning and Machine Learning on the Semantic Web*, 2009, S. 13
- [NMW10] NAGY, Robert ; MEYER-WEGENER, Klaus: Towards Extensible Automatic Image Annotation with the Bag-of-Words Approach. In: *ACM International Workshop on Very-Large-Scale Multimedia Corpus, Mining and Retrieval*, 2010, S. 43–48. <http://dx.doi.org/10.1145/1878137.1878148>
- [NNM96a] NENE, Sameer A. ; NAYAR, Shree K. ; MURASE, Hiroshi: Columbia Object Image Library (COIL-100) / Department of Computer Science, Columbia University, New York. 1996 (CUCS-006-96). – Forschungsbericht
- [NNM96b] NENE, Sameer A. ; NAYAR, Shree K. ; MURASE, Hiroshi: Columbia Object Image Library (COIL-20) / Department of Computer Science, Columbia University, New York. 1996 (CUCS-005-96). – Forschungsbericht
- [NR03] NORVIG, Peter ; RUSSELL, Stuart: *Artificial Intelligence - A Modern Approach*. 2. Prentice Hall Inc., 2003
- [NS06] NISTÉR, David ; STEWÉNIUS, Henrik: Scalable Recognition with a Vocabulary Tree. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 2, 2006, S. 2161–2168. <http://dx.doi.org/10.1109/CVPR.2006.264>

- [NST⁺06] NAPHADE, Milind ; SMITH, John R. ; TESIC, Jelena ; CHANG, Shih-Fu ; HSU, Winston ; KENNEDY, Lyndon ; HAUPTMANN, Alexander ; CURTIS, Jon: Large-Scale Concept Ontology for Multimedia. In: *IEEE Multimedia* 13 (2006), Nr. 3, S. 86–91. <http://dx.doi.org/10.1109/MMUL.2006.63>
- [NW70] NEEDLEMAN, Saul B. ; WUNSCH, Christian D.: A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins. In: *Journal of Molecular Biology* 48 (1970), Nr. 3, S. 443–453. [http://dx.doi.org/10.1016/0022-2836\(70\)90057-4](http://dx.doi.org/10.1016/0022-2836(70)90057-4)
- [OM05] OBDRZÁLEK, Stepán ; MATAS, Jirí: Sub-linear Indexing for Large-Scale Object Recognition. In: *British Machine Vision Conference* Bd. 1, 2005, S. 1–10
- [OPFA06] OPELT, Adndreas ; PINZ, Axel ; FUSSENEGGER, Michael ; AUER, Peter: Generic Object Recognition with Boosting. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (2006), Nr. 3, S. 416–431. <http://dx.doi.org/10.1109/TPAMI.2006.54>
- [Orn95] ORNAGER, Susanne: The newspaper image database: empirical supported analysis of users' typology and word association clusters. In: *ACM SIGIR International Conference on Research and Development in Information Retrieval*, 1995, S. 212–218. <http://dx.doi.org/10.1145/215206.215362>
- [OSO03] OZMUTLU, Seda ; SPINK, Amanda ; OZMUTLU, Huseyin C.: Multimedia web searching trends: 1997-2001. In: *Information Processing & Management* 39 (2003), Nr. 4, S. 611–622. [http://dx.doi.org/10.1016/S0306-4573\(02\)00038-9](http://dx.doi.org/10.1016/S0306-4573(02)00038-9)
- [OT01] OLIVA, Aude ; TORRALBA, Antonio: Modeling the shape of the scene: a holistic representation of the spatial envelope. In: *International Journal of Computer Vision* 42 (2001), Nr. 3, S. 145–175. <http://dx.doi.org/10.1023/A:1011139631724>
- [OT06] *Kapitel* Building the gist of a scene: the role of global image features in recognition. In: OLIVA, Aude ; TORRALBA, Antonio: *Progress in Brain Research, Visual Perception - Part 2, Fundamentals of Awareness, Multi-Sensory Integration and High-Order Perception*. Bd. 155. Elsevier, 2006, S. 23–36. [http://dx.doi.org/10.1016/S0079-6123\(06\)55002-2](http://dx.doi.org/10.1016/S0079-6123(06)55002-2)

- [Pan57] PANOFSKY, Erwin: *Meaning in the visual arts.* Garden City, N.Y., Doubleday, 1957
- [PCI⁺07] PHILBIN, James ; CHUM, Ondrej ; ISARD, Michael ; SIVIC, Josef ; ZISSELMAN, Andrew: Object Retrieval with Large Vocabularies and Fast Spatial Matching. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2007, S. 1–8. <http://dx.doi.org/10.1109/CVPR.2007.383172>
- [PCST00] PLATT, John C. ; CRISTIANINI, Nello ; SHawe-Taylor, John: Large Margin DAGs for Multiclass Classification. In: *Advances in Neural Information Processing Systems*, 2000, S. 547–553
- [PD07] PERRONNIN, Florent ; DANCE, Christopher: Fisher Kernels on Visual Vocabularies for Image Categorization. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2007, S. 1–8. <http://dx.doi.org/10.1109/CVPR.2007.383266>
- [PDCB06] PERRONNIN, Florent ; DANCE, Christopher ; CSURKA, Gabriela ; BRESSAN, Marco: Adapted Vocabularies for Generic Visual Categorization. In: *European Conference on Computer Vision*, 2006, S. 464–475. <http://dx.doi.org/10.1007/11744085>
- [Per08] PERRONNIN, Florent: Universal and Adapted Vocabularies for Generic Visual Categorization. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (2008), Nr. 7, S. 1243–1256. <http://dx.doi.org/10.1109/TPAMI.2007.70755>
- [PHD05] PETRIE, Helen ; HARRISON, Chandra ; DEV, Sundeep: Describing images on the Web: a survey of current practice and prospects for the future. In: *Human Computer Interaction International (HCII)*, 2005
- [PJA10] PAULEVÉ, Loic ; JÉGOU, Hervé ; AMSALEG, Laurent: Locality Sensitive Hashing: A Comparison of Hash Function Types and Querying Mechanisms. In: *Pattern Recognition Letters* 31 (2010), S. 1348–1358. <http://dx.doi.org/10.1016/j.patrec.2010.04.004>
- [PMHGP11] PREKOPCSÁK, Zoltán ; MAKRAI, Gábor ; HENK, Tamás ; GÁSPÁR-PAPANEK, Csaba: Radoop: Analyzing Big Data with RapidMiner and Hadoop. In: *RapidMiner Community Meeting And Conference*, 2011

- [Pro11] PROTSENKO, Sergiy: *Entwurf un Implementierung einer Annotationschnittstelle für Pixtract*, Friedrich-Alexander-Universität Erlangen-Nürnberg, Diplomarbeit, 2011
- [PSM10] PERRONNIN, Florent ; SÁNCHEZ, Jorge ; MENSINK, Thomas: Improving the Fisher Kernel for Large-Scale Image Classification. In: *European Conference on Computer Vision*, 2010, S. 143–156. http://dx.doi.org/10.1007/978-3-642-15561-1_11
- [Pu05] PU, Hsiao-Tieh: A comparative analysis of web image and textual queries. In: *Online Information Review* 29 (2005), Nr. 5, S. 457–467. <http://dx.doi.org/10.1108/14684520510628864>
- [Pu08] PU, Hsiao-Tieh: An analysis of failed queries for web image retrieval. In: *Journal of Information Science* 34 (2008), Nr. 3, S. 275–289. <http://dx.doi.org/10.1177/0165551507084140>
- [PVS08] PEREIRA, Fernando ; VETRO, Anthony ; SIKORA, Thomas: Multimedia Retrieval and Delivery: Essential Metadata Challenges and Standards. In: *Proceedings of the IEEE* 96 (2008), Nr. 4, S. 721–744. <http://dx.doi.org/10.1109/JPROC.2008.916384>
- [PYDF04] PAN, Jia-Yu ; YANG, Hyung-Jeong ; DUYGULU, Pinar ; FALOUTSOS, Christos: Automatic Image Captioning. In: *IEEE International Conference on Multimedia and Expo* Bd. 3, 2004, S. 1987–1990. <http://dx.doi.org/10.1109/ICME.2004.1394652>
- [QT09] QUATTTONI, Ariadna ; TORRALBA, Antonio: Recognizing Indoor Scenes. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2009, S. 413–420. <http://dx.doi.org/10.1109/CVPR.2009.5206537>
- [Raa93] RAASCH, Jörg: *Systementwicklung mit strukturierten Methoden*. 3. Hanser, 1993
- [Rah96] RAHM, Erhard: *Mehrrechner-Datenbanksysteme*. Addison-Wesley, 1996
- [Res95] RESNIK, Philip: Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In: *AAAI International Joint Conference on Artificial Intelligence*, 1995, S. 448–453
- [Rij76] RIJSBERGEN, C.J. van: *Information Retrieval*. Butterworth, 1976

- [RK04] RIFKIN, Ryan ; KLAUTAU, Aldebaro: In Defense of One-Vs-All Classification. In: *Journal of Machine Learning Research* 5 (2004), S. 101–141
- [RLG⁺05] Kapitel Mining Time Series Data. In: RATANAMAHATANA, Chotirat A. ; LIN, Jessica ; GUNOPULOS, Dimitrios ; KEOGH, Eamonn ; VLACHOS, Michail ; DAS, Gautam: *The Data Mining and Knowledge Discovery Handbook*. Springer, 2005, S. 1069–1103. http://dx.doi.org/10.1007/0-387-25465-X_51
- [RPE⁺04] RESNIKOFF, Serge ; PASCOLINI, Donatella ; ETYA’ALE, Daniel ; KOCUR, Ivo ; PARARAJASEGARAM, Ramachandra ; POKHAREL, Gopal P. ; MARIOTTI, Silvio P.: Global Data on Visual Impairment in the Year 2002. In: *Bulletin of the World Health Organization* 82 (2004), Nr. 11, S. 844–851
- [RTL⁺07] RUSSELL, Bryan C. ; TORRALBA, Antonio ; LIU, Ce ; FERGUS, Rob ; FREEMAN, William T.: Object Recognition by Scene Alignment. In: *Advances in Neural Information Processing Systems*, 2007
- [RTMF08] RUSSELL, Bryan C. ; TORRALBA, Antonio ; MURPHY, Kevin P. ; FREEMAN, William T.: LabelMe: A Database and Web-Based Tool for Image Annotation. In: *International Journal of Computer Vision* 77 (2008), S. 157–173. <http://dx.doi.org/10.1007/s11263-007-0090-8>
- [Sam06] SAMET, Hanan ; SAMET, Hanan (Hrsg.): *Foundations of Multidimensional and Metric Data Structures*. Morgan Kaufmann, 2006. – ISBN 978-0-12-369446-1
- [Sap11] SAPIRO, Guillermo: Images Everywhere: Looking for Models - Technical Perspective. In: *Communications of the ACM* 54 (2011), Nr. 5, S. 108–108. <http://dx.doi.org/10.1145/1941487.1941512>
- [Sav11] SAVAGE, Neil: Sorting Through Photos. In: *Communications of the ACM* 54 (2011), Nr. 5, S. 13–15. <http://dx.doi.org/10.1145/1941487.1941493>
- [SB91] SWAIN, Michael J. ; BALLARD, Dana H.: Color Indexing. In: *International Journal of Computer Vision* 7 (1991), Nr. 1, S. 11–32. <http://dx.doi.org/10.1007/BF00130487>
- [SBB⁺09] STEGMAIER, Florian ; BAILER, Werner ; BÜRGGER, Tobias ; DÖLLER, Mario ; HÖFFERNIG, Martin ; LEE, Wonsuk ; MALAISÉ, Véronique ;

- POPPE, Chris ; TRONCY, Raphaël ; KOSCH, Harald ; WALLE, Rik Van d.: How to Align Media Metadata Schemas? Design and Implementation of the Media Ontology. In: *Workshop on Semantic Multimedia Database Technologies, CEUR Workshop Proceedings* Bd. 539, 2009
- [SBLB04] SHENA, Xipeng ; BOUTELLA, Matthew ; LUOB, Jiebo ; BROWNA, Christopher: Multi-label machine learning and its application to semantic scene classification. In: *IS&T/SPIE's Sixteenth Annual Symposium on Electronic Imaging: Science and Technology*, 2004, S. 188–199. <http://dx.doi.org/10.1117/12.523428>
- [SBV95] SCHÖLKOPF, Bernhard ; BURGES, Chris ; VAPNIK, Vladimir N.: Extracting Support Vectors for a Given Task. In: *International Conference on Knowledge Discovery and Data Mining*, 1995, S. 252–257
- [SDRL06] SCHLICKER, Andreas ; DOMINGUES, Francisco S. ; RAHNENFÜHRER, Jörg ; LENGAUER, Thomas: A new measure for functional similarity of gene products based on Gene Ontology. In: *BMC Bioinformatics* 7 (2006), Nr. 1, S. 302. <http://dx.doi.org/10.1186/1471-2105-7-302>
- [SGS08] SANDE, Koen E. A. d. ; GEVERS, Theo ; SNOEK, Cees G. M.: Evaluation of Color Descriptors for Object and Scene Recognition. In: *IEEE International Conference on Computer Vision and Pattern Recognition*. Anchorage, Alaska, USA, June 2008, S. 1–8. <http://dx.doi.org/10.1109/CVPR.2008.4587658>
- [SGS10] SANDE, Koen E. A. d. ; GEVERS, Theo ; SNOEK, Cees G. M.: Evaluating Color Descriptors for Object and Scene Recognition. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32 (2010), Nr. 9, S. 1582–1596. <http://dx.doi.org/10.1109/TPAMI.2009.154>
- [SGTS07] STVILIA, Besiki ; GASSER, Les ; TWIDALE, Michael B. ; SMITH, Linda C.: A Framework for Information Quality Assessment. In: *Journal of the American Society for Information Science and Technology* 58 (2007), Nr. 12, S. 1720–1733. <http://dx.doi.org/10.1002/asi.20652>
- [Shn92] SHNEIDERMAN, Ben: Tree visualization with tree-maps: 2-d space-filling approach. In: *ACM Transactions on Graphics* 11 (1992), Nr. 1, S. 92–99. <http://dx.doi.org/10.1145/102377.115768>

- [SJWS02] SPINK, Amanda ; JANSEN, Bernard J. ; WOLFRAM, Dietmar ; SARASEVIC, Tefko: From E-Sex to E-Commerce: Web Search Changes. In: *IEEE Computer* 35 (2002), Nr. 3, S. 107–109. <http://dx.doi.org/10.1109/2.989940>
- [SL94] SHATFORD LAYNE, Sarah: Some Issues in the Indexing of Images. In: *Journal of the American Society for Information Science and Technology* 45 (1994), Nr. 8, S. 583–588. [http://dx.doi.org/10.1002/\(SICI\)1097-4571\(199409\)45:8<583::AID-ASI13>3.0.CO;2-N](http://dx.doi.org/10.1002/(SICI)1097-4571(199409)45:8<583::AID-ASI13>3.0.CO;2-N)
- [SL01] SUN, Aixin ; LIM, Ee-Peng: Hierarchical text classification and evaluation. In: *IEEE International Conference on Data Mining*, 2001, S. 521–528. <http://dx.doi.org/10.1109/ICDM.2001.989560>
- [SMB05] SCANNAPIECO, Monica ; MISSIER, Paolo ; BATINI, Carlo: Data Quality at a Glance. In: *Datenbank-Spektrum* 14 (2005), S. 6–14
- [SP94] SRINIVAS, M. ; PATNAIK, Lalit M.: Genetic Algorithms: A Survey. In: *IEEE Computer* 27 (1994), Nr. 6, S. 17–26. <http://dx.doi.org/10.1109/2.294849>
- [SRE⁺05] SIVIC, Josef ; RUSSELL, Bryan C. ; EFROS, Alexei A. ; ZISSERMAN, Andrew ; FREEMAN, William T.: Discovering Objects and Their Location in Images. In: *IEEE International Conference on Computer Vision*, 2005, S. 370–377. <http://dx.doi.org/10.1109/ICCV.2005.77>
- [SRSS06] SONNENBURG, Sören ; RÄTSCH, Gunnar ; SCHÄFER, Christin ; SCHÖLKOPF, Bernhard: Large Scale Multiple Kernel Learning. In: *Journal of Machine Learning Research* 7 (2006), S. 1531–1565
- [SS01] SHAPIRO, Linda G. ; STOCKMAN, George C.: *Computer Vision*. Prentice Hall Inc., 2001
- [SS04] SCHAEFER, Gerald ; STICH, Michal: UCID - An Uncompressed Colour Image Database. In: *Proceedings of SPIE, Storage and Retrieval Methods and Applications for Multimedia*, 2004, S. 472–480. <http://dx.doi.org/10.1117/12.525375>
- [SS06] SMITH, John R. ; SCHIRLING, Peter: Metadata Standards Roundup. In: *IEEE Multimedia* 13 (2006), Nr. 2, S. 84–88. <http://dx.doi.org/10.1109/MMUL.2006.34>

- [SS11] SCHNELLE-SCHNEYDER, Marlene: *Sehen und Photographie – Ästhetik und Bild.* 2. X.media.press, Springer, 2011. <http://dx.doi.org/10.1007/978-3-642-15150-7>
- [ST09] SHINOHARA, Kirsten ; TENENBERG, Josh: A Blind Person's Interactions with Technology. In: *Communications of the ACM* 52 (2009), Nr. 8, S. 58–66. <http://dx.doi.org/10.1145/1536616.1536636>
- [Sti06] STILL, Michael: *The Definitive Guide to ImageMagick.* Apress, 2006
- [SVBM05] SRIKANTH, Munirathnam ; VARNER, Joshua ; BOWDEN, Mitchell ; MOLDOVAN, Dan: Exploiting ontologies for automatic image annotation. In: *ACM SIGIR International Conference on Research and Development in Information Retrieval*, 2005, S. 552–558. <http://dx.doi.org/10.1145/1076034.1076128>
- [SW78] SALTON, Gerard M. ; WONG, Andrew K. C.: Generation and Search of Clustered Files. In: *ACM Transactions on Database Systems* 3 (1978), Nr. 4, S. 321–346. <http://dx.doi.org/10.1145/320289.320291>
- [SZ02] SCHAFFALITZKY, Frederik ; ZISSEMAN, Andrew: Multi-view Matching for Unordered Image Sets, or „How Do I Organize My Holiday Snaps?“. In: *European Conference on Computer Vision*, 2002, S. 414–431. http://dx.doi.org/10.1007/3-540-47969-4_28
- [Sze11] SZELISKI, Richard: *Computer Vision - Algorithms and Applications.* Springer, 2011
- [SZF07] SCHICKEL-ZUBER, Vincent ; FALTINGS, Boi: OSS: A Semantic Similarity Function based on Hierarchical Ontologies. In: *AAAI International Joint Conference on Artificial Intelligence*, 2007, S. 551–556
- [SZR00] SRIHARI, Rohini K. ; ZHANG, Zhongfei ; RAO, Aibing: Intelligent Indexing and Semantic Retrieval of Multimodal Documents. In: *Information Retrieval* 2 (2000), Nr. 2-3, S. 245–275. <http://dx.doi.org/10.1023/A:1009962928226>
- [TA00] TAKAGI, Hironobu ; ASAOKAWA, Chieko: Transcoding proxy for nonvisual web access. In: *ACM International Conference on Assistive Technologies*, 2000, S. 164–171. <http://dx.doi.org/10.1145/354324.354371>

- [Tes11] *Tesseract*. <http://code.google.com/p/tesseract-ocr/>. Version: 22.01. 2011
- [TFW08] TORRALBA, Antonio ; FERGUS, Rob ; WEISS, Yair: Small Codes and Large Image Databases for Recognition. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2008, S. 1–8. <http://dx.doi.org/10.1109/CVPR.2008.4587633>
- [TG04] TUYTELAARS, Tinne ; GOOL, Luc van: Matching Widely Separated Views Based on Affine Invariant Regions. In: *International Journal of Computer Vision* 59 (2004), Nr. 1, S. 61–85. <http://dx.doi.org/10.1023/B:VISI.0000020671.28016.e8>
- [Thy] THYSSEN, Anthony: *Perspective Projection Distortion*. http://www.imagemagick.org/Usage/distorts/#perspective_projection. – Online-Ressource, Abruf: 21.01.2011
- [TIF92] ; Adobe Systems Incorporated (Veranst.): *TIFF Revision 6.0*. <http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf>. Version: June 1992
- [TJBB06] TEH, Yee W. ; JORDAN, Michael I. ; BEAL, Matthew J. ; BLEI, David M.: Hierarchical Dirichlet Processes. In: *Journal of the American Statistical Association* 101 (2006), Nr. 476, S. 1566–1581. <http://dx.doi.org/10.1198/016214506000000302>
- [TM08] TUYTELAARS, Tinne ; MIKOŁAJCZYK, Krystian: *Local Invariant Feature Detectors: A Survey*. Now Publishers, 2008. – 177–280 S. <http://dx.doi.org/10.1561/0600000017>
- [TMF04] TORRALBA, Antonio ; MURPHY, Kevin P. ; FREEMAN, William T.: Sharing Features: Efficient Boosting Procedures for Multiclass Object Detection. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 2, 2004, S. 762–769. <http://dx.doi.org/10.1109/CVPR.2004.1315241>
- [Tor03] TORRALBA, Antonio: Contextual Priming for Object Detection. In: *International Journal of Computer Vision* 53 (2003), Nr. 2, S. 169–191. <http://dx.doi.org/10.1023/A:1023052124951>

- [Tor09] TORRALBA, Antonio: How Many Pixels Make an Image? In: *Visual Neuroscience* 26 (2009), S. 123–131. <http://dx.doi.org/10.1017/S0952523808080930>
- [TSJ09] TJONDRONEGORO, Dian ; SPINK, Amanda ; JANSEN, Bernard J.: A Study and Comparison of Multimedia Web Searching: 1997-2006. In: *Journal of the American Society for Information Science and Technology* 60 (2009), Nr. 9, S. 1756–1768. <http://dx.doi.org/10.1002/asi.21094>
- [TSU⁺08] TAHIR, Muhammad A. ; SANDE, Koen van d. ; UIJLINGS, Jasper ; YAN, Fei ; LI, Xirong ; MIKOLAJCZYK, Krystian ; KITTLER, Josef ; GEVERS, Theo ; SMEULDERS, Arnold: *SurreyUVA SRK-DA method*. <http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2008/workshop/tahir.pdf>. Version: 2008
- [UB05] ULUSOY, Ilkay ; BISHOP, Christopher M.: Generative Versus Discriminative Methods for Object Recognition. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 2, 2005, S. 258–265. <http://dx.doi.org/10.1109/CVPR.2005.167>
- [UB06] Kapitel 9. In: ULUSOY, Ilkay ; BISHOP, Christopher M.: *Lecture Notes in Computer Science*. Bd. 4170: *Comparison of Generative and Discriminative Techniques for Object Detection and Classification*. Springer, 2006. – ISBN 3-540-68794-7, S. 173–195. http://dx.doi.org/10.1007/11957959_9
- [Uhl10] UHL, Alexander: *Webschnittstelle zur Verwaltung und Visualisierung von Objektkategorien*, Friedrich-Alexander-Universität Erlangen-Nürnberg, Diplomarbeit, 2010
- [Vap95] VAPNIK, Vladimir N.: *The Nature of Statistical Learning Theory*. Springer, 1995
- [VJ01] VIOLA, Paul ; JONES, Michael: Rapid Object Detection using a Boosted Cascade of Simple Features. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 1, 2001, S. 511–518. <http://dx.doi.org/10.1109/CVPR.2001.990517>
- [VR07] VARMA, Manik ; RAY, Debajyoti: Learning the Discriminative Power-Invariance Trade-Off. In: *IEEE International Conference on Computer Vision*, 2007, S. 1–8. <http://dx.doi.org/10.1109/ICCV.2007.4408875>

- [VWS01] VENDRIG, Jeroen ; WORRING, Marcel ; SMEULDERS, Arnold W. M.: Filter Image Browsing: Interactive Image Retrieval by Using Database Overviews. In: *Multimedia Tools and Applications* 15 (2001), Nr. 1, S. 83–103. <http://dx.doi.org/10.1023/A:1011367820253>
- [WAC⁺04] WILLAMOWSKI, Jutta ; ARREGUI, Damian ; CSURKA, Gabriella ; DANCE, Chris ; FAN, Lixin: Categorizing Nine Visual Classes using Local Appearance Descriptors. In: *IEEE International Conference on Pattern Recognition, Learning for Adaptable Visual Systems Workshop*, 2004
- [WB10] WANG, Kai ; BELONGIE, Serge: Word Spotting in the Wild. In: *European Conference of Computer Vision*, 2010, S. 591–604. http://dx.doi.org/10.1007/978-3-642-15549-9_43
- [WCA] *Web Content Accessibility Guidelines*. <http://www.w3.org/WAI/intro/wcag.php>
- [WCY04] WU, Wen ; CHEN, Xilin ; YANG, Jie: Incremental detection of text on road signs from video with application to a driving assistant system. In: *ACM International Conference on Multimedia*, 2004, S. 852–859. <http://dx.doi.org/10.1145/1027527.1027724>
- [WDJ11] WENGERT, Christian ; DOUZE, Matthijs ; JÉGOU, Hervé: Bag-of-colors for Improved Image Search. In: *ACM International Conference on Multimedia*, 2011, S. 1437–1440. <http://dx.doi.org/10.1145/2072298.2072034>
- [WDS⁺01] WENYIN, Liu ; DUMAIS, Susan ; SUN, Yanfeng ; ZHANG, HongJiang ; CZERWINSKI, Mary ; FIELD, Brent: Semi-Automatic Image Annotation. In: *Human-Computer Interaction International (HCII)*, 2001, S. 326–333
- [Whi10] WHITE, Tom ; WHITE, Tom (Hrsg.): *Hadoop: The Definitive Guide*. 2. O'Reilly Media, 2010. – ISBN 978–1449389734
- [Win90] WINKLER, William E.: String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. In: *Section on Survey Research Methods*, 1990, S. 354–359
- [WJZZ06] WANG, Changhu ; JING, Feng ; ZHANG, Lei ; ZHANG, Hong-Jiang: Image Annotation Refinement using Random Walk with Restarts. In: *ACM International Conference on Multimedia*, 2006, S. 647–650. <http://dx.doi.org/10.1145/1180639.1180774>

- [WJZZ07] WANG, Changhu ; JING, Feng ; ZHANG, Lei ; ZHANG, Hong-Jiang: Content-Based Image Annotation Refinement. In: *IEEE International Conference on Computer Vision and Pattern Recognition*, 2007, S. 1–8. <http://dx.doi.org/10.1109/CVPR.2007.383221>
- [WJZZ08] WANG, Changhu ; JING, Feng ; ZHANG, Lei ; ZHANG, Hong-Jiang: Scalable search-based image annotation. In: *Multimedia Systems* 14 (2008), Nr. 4, S. 205–220. <http://dx.doi.org/10.1007/s00530-008-0128-y>
- [Wöl11] WÖLFL, Andreas: *Optimierung und parallele Ausführung von similarity-based image multi-joins*, Universität Passau, Fakultät für Informatik und Mathematik, Diplomarbeit, Juni 2011
- [WLMH09] WEINMAN, Jerod J. ; LEARNED-MILLER, Erik ; HANSON, Allen R.: Scene Text Recognition Using Similarity and a Lexicon with Sparse Belief Propagation. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31 (2009), Nr. 10, S. 1733–1746. <http://dx.doi.org/10.1109/TPAMI.2009.38>
- [Wol94] WOLBERG, George: *Digital Image Warping*. Los Alamitos, CA, USA : IEEE Computer Society Press, 1994. – ISBN 0818689447
- [WP94] WU, Zhibiao ; PALMER, Martha: Verbs semantics and lexical selection. In: *ACM Annual Meeting of the Association for Computer Linguistics*, 1994, S. 133–138. <http://dx.doi.org/10.3115/981732.981751>
- [WSB98] WEBER, Roger ; SCHEK, Hans-J. ; BLOTT, Stephen: A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces. In: *International Conference on Very Large Databases*, 1998, S. 194–205
- [WTF08] WEISS, Yair ; TORRALBA, Antonio ; FERGUS, Rob: Spectral Hashing. In: *Advances in Neural Information Processing Systems*, 2008
- [WZFF06] WANG, Gang ; ZHANG, Ye ; FEI-FEI, Li: Using Dependent Regions for Object Categorization in a Generative Framework. In: *IEEE International Conference on Computer Vision and Pattern Recognition* Bd. 2, 2006, S. 1597–1604. <http://dx.doi.org/10.1109/CVPR.2006.324>
- [WZJM06a] WANG, Xin-Jing ; ZHANG, Lei ; JING, Feng ; MA, Wei-Ying: AnnoSearch: Image Auto-Annotation by Search. In: *IEEE International Conference*

- on Computer Vision and Pattern Recognition Bd. 2, 2006, S. 1483–1490. <http://dx.doi.org/10.1109/CVPR.2006.58>
- [WZJM06b] WANG, Xin-Jing ; ZHANG, Lei ; JING, Feng ; MA, Wei-Ying: Image annotation using search and mining technologies. In: *ACM International Conference on World Wide Web*, 2006, S. 1045–1046. <http://dx.doi.org/10.1145/1135777.1136007>
- [XMP08a] ; Adobe Systems Incorporated (Veranst.): *Extensible Metadata Platform (XMP) Specification: Part 1, Data and Serialization Model*. <http://www.adobe.com/devnet/xmp/pdfs/XMPSpecificationPart1.pdf>. Version: 2008
- [XMP08b] ; Adobe Systems Incorporated (Veranst.): *Extensible Metadata Platform (XMP) Specification: Part 2, Standard Schemas*. <http://www.adobe.com/devnet/xmp/pdfs/XMPSpecificationPart2.pdf>. Version: 2008
- [XMP08c] ; Adobe Systems Incorporated (Veranst.): *Extensible Metadata Platform (XMP) Specification: Part 3, Storage in Files*. <http://www.adobe.com/devnet/xmp/pdfs/XMPSpecificationPart3.pdf>. Version: 2008
- [YJHN07] YANG, Jun ; JIANG, Yu-Gang ; HAUPTMANN, Alexander G. ; NGO, Chong-Wah: Evaluating bag-of-visual-words representations in scene classification. In: *International Workshop on Multimedia Information Retrieval*, 2007, S. 197–206. <http://dx.doi.org/10.1145/1290082.1290111>
- [YLM⁺06] YUAN, Xun ; LAI, Wei ; MEI, Tao ; HUA, Xian-Sheng ; WU, Xiu-Qing ; LI, Shipeng: Automatic Video Genre Categorization using Hierarchical SVM. In: *IEEE International Conference on Image Processing*, 2006, S. 2905–2908. <http://dx.doi.org/10.1109/ICIP.2006.313037>
- [YSR05] YAVLINSKY, Alexei ; SCHOFIELD, Edward ; RÜGER, Stefan: Automated Image Annotation Using Global Features and Robust Nonparametric Density Estimation. In: *ACM International Conference on Image and Video Retrieval*, 2005, S. 507–517. http://dx.doi.org/10.1007/11526346_54
- [YYZL03] YANG, Jian ; YANG, Jing-yu ; ZHANG, David ; LU, Jian-feng: Feature fusion: parallel strategy vs. serial strategy. In: *Pattern Recognition* 36 (2003), Nr. 6, S. 1369–1381. [http://dx.doi.org/10.1016/S0031-3203\(02\)00262-5](http://dx.doi.org/10.1016/S0031-3203(02)00262-5)

- [ZCJ05] ZHANG, Chen ; CHAI, Joyce Y. ; JIN, Rong: User term feedback in interactive text-based image retrieval. In: *ACM SIGIR International Conference on Research and Development in Information Retrieval*, 2005, S. 51–58. <http://dx.doi.org/10.1145/1076034.1076046>
- [ZMLS07] ZHANG, Jianguo ; MARSZAŁEK, Marcin ; LAZEBNIK, Svetlana ; SCHMID, Cordelia: Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study. In: *International Journal of Computer Vision* 73 (2007), Nr. 2, S. 213–238. <http://dx.doi.org/10.1007/s11263-006-9794-4>
- [ZMP04] ZELNIK-MANOR, Lihi ; PERONA, Pietro: Self-Tuning Spectral Clustering. In: *Advances in Neural Information Processing Systems*, 2004, S. 1601–1608
- [ZW07] ZWEIG, Alon ; WEINSHALL, Daphna: Exploiting Object Hierarchy: Combining Models from Different Category Levels. In: *IEEE International Conference on Computer Vision*, 2007, S. 1–8. <http://dx.doi.org/10.1109/ICCV.2007.4409064>
- [ZWQ⁺05] ZHIGANG, Liu ; WENZHONG, Shi ; QIANQING, Qin ; XIAOWEN, Li ; DONGHUI, Xie: Hierarchical Support Vector Machines. In: *IEEE International Geoscience and Remote Sensing Symposium* Bd. 1, 2005, S. 4. <http://dx.doi.org/10.1109/IGARSS.2005.1526138>
- [ZYC06] ZHU, Qiang ; YEH, Mei-Chen ; CHENG, Kwang-Ting: Multimodal fusion using learned text concepts for image categorization. In: *ACM International Conference on Multimedia*, 2006, S. 211–220. <http://dx.doi.org/10.1145/1180639.1180698>