



Hochschule für Technik,
Wirtschaft und Kultur Leipzig

Bachelorarbeit
zur Erlangung des akademischen Grades

Bachelor of Science (B.Sc.)

im Bachelorstudiengang Medieninformatik
der Fakultät Informatik und Medien

Entwicklung eines Bilderkennungssystems zum finden von Duplikaten in einer Bilddatenbank bei der Registrierung neuer Bilder

vorgelegt von
Martino Thomann

Leipzig, den 12. September 2023

Erstprüfer: Prof. Dr. Sibylle Schwarz
Zweitprüfer: B.Sc. Marc Bellmann

Zusammenfassung

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Aufgabenstellung	1
1.3	Erfolgs- und Qualitätskriterien	2
2	Grundlagen	3
2.1	Merkmalbasierte Algorithmen	3
2.1.1	SIFT: Scale-Invariant Feature Transform	3
2.1.2	ORB: Oriented FAST and Rotated BRIEF	5
2.2	Test-Metriken	6
3	Analyse	8
4	Formel	9
5	Test-System	10
6	Versuche und Auswertung	11
7	Neues Bilderkennungs-System	12
8	Schluss	I
	Literaturverzeichnis	II
	Abbildungsverzeichnis	III
	Tabellenverzeichnis	IV

1 Einleitung

1.1 Motivation

Bei Spreadshirt werden täglich tausende Designs hochgeladen. Viele davon in der Partner-Area, wo Designer ihre Bilder zum Verkauf anbieten können. Dabei muss seitens Spreadshirt sichergestellt werden, dass die hochgeladenen Designs nicht verfassungsfeindlich sind und auch nicht gegen das Urheberrecht oder Spreadshirts eigene Richtlinien verstoßen. Die manuelle Überprüfung von so vielen Designs ist sehr aufwändig und häufig kommt es vor, dass die gleichen Designs mehrmals hochgeladen werden. Eine automatische Sperrung, von bereits verbotenen Designs, die erneut hochgeladen werden, kann den Überprüfungsprozess entlasten. Daher ist ein System notwendig, dass bei neu hochgeladenen Designs Duplikate in der Datenbank der verbotenen Designs erkennen kann.

1.2 Aufgabenstellung

Ziel dieser Bachelorarbeit ist die Entwicklung eines Systems, dass in der Lage ist, bei neuen Bildern zu erkennen, ob diese bereits in einer Datenbank mit bereits gespeicherten Bildern auftauchen. Ein neues Bild soll auch dann als Duplikat erkannt werden, wenn der Bildinhalt im Vergleich zum Original transformiert, also rotiert, skaliert, verschoben oder gespiegelt wurde. Fälle in denen das Bild anders gefärbt ist oder als Teil eines größeren Bildes auftaucht, sollen ebenfalls berücksichtigt werden.

Zur Implementierung des Systems soll ein "feature-based"(dt. merkmalsbasierter) Algorithmus verwendet werden. Es gibt eine Vielzahl an merkmalsbasierten Algorithmen, die jeweils ihre eigenen Stärken und Schwächen haben. Eine Auswahl dieser Algorithmen sollen innerhalb der Arbeit für den Anwendungsfall bei Spreadshirt getestet und miteinander verglichen werden.

1.3 Erfolgs- und Qualitätskriterien

Die Güte des Bilderkennungssystems soll anhand von Testdatensätzen ermittelt werden. Die Testdatensätze sind dabei in zwei Teilsätzen unterteilt. Der erste Teilsatz an Bildern stellt die Menge an gespeicherten Datenbankbildern dar. Der zweite Teilsatz enthält eine Untermenge an Duplikaten aus dem Datenbanksatz und eine Menge an neuen Bildern, die nicht im Datenbanksatz auftauchen. Getestet wird in verschiedenen Szenarien, die die Robustheit des Systems gegenüber bestimmter Sonderfälle testen soll. Je nach Szenario sind die Duplikate auf unterschiedliche Weise im Vergleich zum Original verändert. Als Vergleich dient der pHash-Algorithmus, der momentan bei Spreadshirt zur Duplikatensuche verwendet wird.

Die verwendeten Metriken werden in den Grundlagen 2.2 erklärt. Am wichtigsten ist dabei ein hoher Recall, sodass möglichst viele Duplikate durch das System abgefangen werden. Da automatisch gesperrte Designs nochmal manuell geprüft werden, fällt eine niedrigere Spezifität bei der Auswertung nicht so sehr ins Gewicht. Laufzeit- und Speicherkosten sollen ebenfalls innerhalb der Arbeit abgeschätzt werden.

Um für die Verwendung bei Spreadshirt in Frage zu kommen, muss das neue System zuverlässiger sein, als die momentan eingesetzte pHash-Implementaion. Dafür wird ein höherer Recall angestrebt. Um sicherzustellen, dass Spezifität nicht zu sehr absinkt, wird auf eine höhere oder zumindest gleichbleibende Balancierte-Genauigkeit abgezielt.

2 Grundlagen

2.1 Merkmalbasierte Algorithmen

Ziel der "feature-based" (dt. Merkmalbasierten) Algorithmen ist es markante Punkte innerhalb von Bildern zu finden und zu beschreiben. Die Algorithmen bestehen dabei im Grunde aus zwei Schritten:

1. Schlüsselpunktsuche, bei der nach Koordinaten innerhalb eines Bildes gesucht wird, an denen sich markante Punkte befinden.
2. Erstellung von Deskriptoren, die den Bereich um die Schlüsselpunkte beschreiben. Diese sollen später mit Deskriptoren aus anderen Bildern verglichen werden, um gemeinsame Merkmale zu finden.

Wie genau diese beiden Schritte implementiert sind ist je nach Algorithmus unterschiedlich.

Dabei soll sichergestellt werden, dass die gefundenen Merkmale robust gegenüber Rotation, Verschiebung und Skalierung sind. Auch der Einfluss durch Bildrauschen und Verzerrung soll möglichst gering sein. [Low99, S. 2]

Häufig handelt es sich bei den gefundenen Merkmalen um Rand- und Eckpunkte oder Details auf einer Fläche.

2.1.1 SIFT: Scale-Invariant Feature Transform

1. Suche nach Extrempunkten SIFT sucht nach Extrempunkten innerhalb von Bildern. Extrempunkte sind dabei Punkte, an denen lokale Maxima oder Minima in der Helligkeit auftreten. Also helle Punkte innerhalb von dunklen Bereichen und dunkle Punkte innerhalb von hellen Bereichen.

Um Extrempunkte zu finden wird der "Difference of Gaussian" genutzt. Zuerst wird eine bestimmte Anzahl an Bildern, durch das Weichzeichnen des Originalbilds mit dem Gaußschen Weichzeichner, erstellt. Der Grad der Unschärfe steigt dabei bei jedem Bild um einen konstanten Faktor. Ein Satz dieser weichgezeichneten Bilder wird dabei als Oktave bezeichnet. Für jedes paar an aneinander grenzenden weichgezeichneten Bildern, wird die Differenz berechnet. Diese Differenz wird als "Difference of Gaussian" bezeichnet. Bei der Implementation von SIFT werden pro Oktave meist fünf weichgezeichnete Bilder verwendet, aus denen vier "Difference of Gaussian" Bilder entstehen. [Low04, S. 94]

Jeder Pixel der mittleren "Difference of Gaussian" Bilder wird mit seinen acht Nachbarn innerhalb desselben Bildes und seinen je neun Nachbarn im DOG darüber und darunter verglichen. Ein Pixel ist dann ein Schlüsselpunkt, wenn sein Helligkeitswert größer oder kleiner als der Wert aller seiner Nachbarn ist. [Low04, S. 95]

Diese Extrempunktsuche wird dabei für mehrere Oktaven durchgeführt. Dabei wird für jede Oktave das Originalbild um den Faktor 2 herunter skaliert. Das ist notwendig, da manche Extrempunkte nur bei bestimmten Auflösungen auffindbar sind. [Low04, S. 94f] Durch die Sammlung von Schlüsselpunkten auf verschiedenen Auflösungen, sind diese bei Bildern in verschiedenen Auflösungen vergleichbar. Der Aufbau von "Difference of Gaussian" Bildern innerhalb von mehreren Oktaven wird als scale-space (dt. Bildpyramide) bezeichnet.

2. Akkurate Ortsbestimmung der Schlüsselpunkte Die Positionen der Schlüsselpunkte, die auf niedrigen Auflösungen gefunden wurden, entspricht gegebenenfalls nicht genau der Position in der Originalauflösung. Deshalb wird die Taylorreihe verwendet, um die tatsächliche Position zu interpolieren. [Low04, S. 97f]

Außerdem werden Schlüsselpunkte an Rändern und Schlüsselpunkte mit niedrigem Kontrast zu ihrer Umgebung herausgefiltert. Das ist nötig, da solche Schlüsselpunkte Deskriptoren hervorbringen, die nicht zuverlässig verglichen werden können. [Low04, 98f]

3. Bestimmung der Schlüsselpunkt Ausrichtung Um Schlüsselpunkte und die aus ihnen resultierenden Deskriptoren robust gegenüber Bildrotationen zu machen, wird jedem Schlüsselpunkt eine Ausrichtung zugewiesen. Damit diese Ausrichtung konsistent ist, wird diese anhand der lokalen Bildeigenschaften um den Schlüsselpunkt herum bestimmt. [Low04, S. 99]

Um das zu erreichen, wird ein 360 Grad umfassendes Histogramm mit 36 bins aufgebaut. Anhand der Skalierung des Schlüsselpunkts wird ein weichgezeichnetes Bild mit passender

Skalierung gewählt. Innerhalb dieses Bildes werden Abtastpunkte um den Schlüsselpunkt herum genommen. Für jeden Abtastpunkt wird die Orientierung und Stärke des Gradienten bestimmt. Der Gradient beschreibt dabei die Änderung der Pixelhelligkeit innerhalb eines Bildes. Sowohl Stärke als auch Orientierung des Gradienten sind für alle Ebenen der Bildpyramide vorberechnet worden. [Low04, S. 99]

Jede Gradient-Orientierung wird in das Histogramm eingetragen und anhand der Gradient-Stärke gewichtet. Da das Histogramm 36 bins hat, werden Orientierungen in 10-Grad-Schritten zusammengefasst. Aus dem Histogramm wird die Orientierung des globalen Maximums, sowie die Orientierungen aller lokalen Maxima die mindestens 80 Prozent des globalen Maximums betragen, entnommen. Für jede entnommene Orientierung wird ein Schlüsselpunkt erstellt. Dadurch kann es mehrere Schlüsselpunkte mit der gleichen Position und Skalierung aber mit unterschiedlichen Orientierungen geben. [Low04, S. 100]

4. Deskriptoren erstellen In dem Bereich um den Schlüsselpunkt werden wie im Schritt 3 (Abschnitt 2.1.1) für jeden Abtastpunkt die Gradienten-Stärke und -Orientierung bestimmt. Der Bereich wird dabei in 4x4 große Unterbereiche aufgeteilt, für die ein Histogramm erstellt wird. Dabei werden für das Histogramm nur 8 bins verwendet, in denen Gradienten-Stärken mit ähnlicher Orientierung aufsummiert werden. Aus diesen Werten wird ein Deskriptor erstellt. Üblicherweise werden Abtastbereiche mit einer Größe von 16x16 verwendet. Daraus ergeben sich insgesamt 16 Unterbereiche der Größe 4x4 mit je 8 Orientierungs-bins. Ein typischer Deskriptor ist ein Vektor mit $16 * 8 = 128$ Werten. [Low04, S. 101]

2.1.2 ORB: Oriented FAST and Rotated BRIEF

ORB basiert auf dem FAST-Algorithmus für die Schlüsselpunktsuche und dem BRIEF-Algorithmus für die Generierung von Deskriptoren

Schlüsselpunktsuche mit FAST Der FAST-Algorithmus sucht nach Eckpunkten in einem Bild. Dazu wird um jeden Pixel im Bild ein Kreis mit 16 Randpixeln gezogen. Der betrachtete Kandidat-Pixel ist genau dann ein Eckpunkt, wenn eine bestimmte Anzahl an aneinander grenzenden Randpixel heller oder dunkler sind als der Kandidat-Pixel plus oder minus einem bestimmten Schwellenwert. [RD06, S. 4] Bei der Implementation in ORB müssen 9 der 16 Randpunkte heller oder dunkler als der Kandidat-Pixel sein, damit der Pixel sich als Randpunkt qualifiziert. [RRKB11, S. 2565]

2.2 Test-Metriken

Das Bilderkennungssystem soll die Bilder in zwei Klassen einordnen: Duplikate und Unikate. Da es in diesem Fall bei der Klassifizierung nur zwei mögliche Klassen gibt, kann man das Bilderkennungssystem als binären Klassifikator bezeichnen.

Die Performance von binären Klassifikatoren kann durch die Werte einer Wahrheitsmatrix quantifiziert werden. Die Wahrheitsmatrix gibt dabei an, wie viele richtige und falsche Entscheidungen das System bei der Klassifikation getroffen hat. [Tha20, S. 170]

System Klass.:	richtig	falsch
Duplikat	Anzahl richtige Duplikate (TP)	Anzahl falsche Duplikate (FP)
Unikat	Anzahl falsche Unikate (FN)	Anzahl richtige Unikate (TN)

TABELLE 2.1: Wahrheitsmatrix für Duplikaten-Suche

Aus den Werten der Wahrheitsmatrix lassen sich weitere Metriken ableiten, die für die Bewertung eines binären Klassifikators nützlich sein können. Interessant für diese Arbeit sind Recall, Spezifität und Balancierte-Genauigkeit.

Der Recall, oder auch true positive rate, gibt das Verhältnis zwischen den Duplikaten, die das System korrekt klassifiziert hat, und allen Duplikaten, die sich in dem Suchdatensatz befinden, an. [Tha20, S. 172]

$$Recall = \frac{TP}{TP + FN}$$

Die Spezifität, oder auch true negative rate, gibt das Verhältnis zwischen den Unikaten, die das System korrekt klassifiziert hat, und allen Unikaten, die sich im Suchdatensatz befinden an. [Tha20, S. 172]

$$Specificity = \frac{TN}{FP + TN}$$

Die Accuracy (dt. Genauigkeit), gibt das Verhältnis zwischen den Bildern, die das System korrekt klassifiziert hat, und allen Bildern im Suchdatensatz an. Sie spiegelt die Fähigkeit des Systems Bilder richtig zu Klassifizieren wieder. [Tha20, S. 171]

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{Recall * P + Specificity * N}{P + N}$$

Allerdings ist die Genauigkeit kein zuverlässiger Wert, wenn man mit unausgeglichenen Datensätzen arbeitet. Ein Datensatz gilt dann als unausgeglichen, wenn einer Klassifikation mehr Elemente angehören, als der anderen Klassifikation. [Tha20, S. 171] In den Testdatensätzen, die für diese Arbeit verwendet werden, befinden sich mehr Unikate als Duplikate. Dieses Ungleichgewicht kann die Genauigkeit stark beeinflussen. So kann zum Beispiel eine gute Spezifität einen schlechten Recall ausgleichen, wenn der Datensatz zum größten Teil aus Unikaten besteht.

Daher kommt bei unbalancierten Datensätzen die *balanced-accuracy* (dt. *Balancierte-Genauigkeit*) zum Einsatz. [Tha20, S. 175]

$$\textit{Balanced} - \textit{Accuracy} = \frac{\textit{Recall} + \textit{Specificity}}{2}$$

Da sie den Durchschnitt aus Recall und Spezifität bildet, werden Unterschiede zwischen den beiden Werten ausgeglichen. Dadurch ist die *Balancierte-Genauigkeit* robust gegenüber unausgeglichenen Datensätzen.

3 Analyse

4 Formel

5 Test-System

6 Versuche und Auswertung

7 Neues Bilderkennungs-System

8 Schluss

Literaturverzeichnis

- [Low99] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.
- [Low04] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110, 2004.
- [RD06] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I 9*, pages 430–443. Springer, 2006.
- [RRKB11] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee, 2011.
- [Tha20] Alaa Tharwat. Classification assessment methods. *Applied computing and informatics*, 17(1):168–192, 2020.

Abbildungsverzeichnis

Tabellenverzeichnis

2.1	Wahrheitsmatrix für Duplikaten-Suche	6
-----	--	---

Selbständigkeitserklärung

Ich erkläre hiermit, dass ich die vorliegende Graduierungsarbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel verfasst habe. Alle den benutzten Quellen wörtlich oder sinngemäß entnommene Stellen sind als solche einzeln kenntlich gemacht.

Diese Arbeit ist bislang keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht worden.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

Leipzig, 12. September 2023

Unterschrift