

Westfälische Wilhelms-Universität Münster

Institut für Informatik

Praktikum *Computer Vision: Camera Trap Challenge* – Wintersemester 2018/19

Dozenten: Jun.-Prof. Dr. Benjamin Risse, Andreas Nienkötter

---

# **Auswertung von Kamerafallenbildern mithilfe von Principal Components Analysis, Spatial Pyramid Matching und Support Vector Machines**

---

Thomas Poschadel  
Rudolf-Harbig-Weg 36, 48149 Münster  
M.Sc. Informatik  
Matrikelnummer: 123 456  
blabla@wwu.de

Joschka Strüber  
Rudolf-Harbig-Weg 36, 48149 Münster  
M.Sc. Informatik  
Matrikelnummer: 418 702  
j.st@wwu.de

Sufian Zaabalawi  
Straße Hausnummer, 12345 Münster  
M.Sc. Informatik  
Matrikelnummer: 123 456  
blabla@wwu.de

Münster, 27. März 2019

# Inhaltsverzeichnis

<b>1 Einleitung</b>	<b>1</b>
<b>2 Aufteilung auf Sequenzen</b>	<b>2</b>
2.1 Sortierung mit Hilfe der EXIF-Daten . . . . .	2
2.2 Camera Trap Sequencer . . . . .	2
<b>3 Lokalisierung mit Principal Components Analysis (PCA)</b>	<b>3</b>
3.1 Hintergrundapproximation mit PCA . . . . .	3
3.2 Sliding-window Lokalisierung mit PCA . . . . .	5
3.3 Objektdetektion mit PCA . . . . .	5
3.3.1 Algorithmus . . . . .	5
<b>4 Klassifizierung mit Histograms of Gradients und Support Vector Machines</b>	<b>7</b>
<b>5 Klassifizierung mit Spatial Pyramid Matching</b>	<b>7</b>
5.1 Grundidee Spatial Pyramid Matching . . . . .	7
5.2 Locality-constrained Linear Coding . . . . .	7
5.3 Implementierung . . . . .	7
<b>6 Evaluierung</b>	<b>7</b>
6.1 Daten . . . . .	7
6.2 Laufzeit Spatial Pyramid Matching . . . . .	7
6.3 Klassifizierung Spatial Pyramid Matching . . . . .	7
<b>7 Fazit</b>	<b>7</b>
<b>Literaturverzeichnis</b>	<b>8</b>
<b>Eigenständigkeitserklärung</b>	<b>8</b>

## 1 Einleitung

Kamerafallen bieten eine immer wichtiger werdende Möglichkeiten Populationen zu überwachen und erforschen. Forschungsinteressen sind beispielsweise die Veränderung der Biodiversität, der Einfluss des Klimawandels und anderer Einflüsse auf die Lebensräume und die Migrationsmuster von Populationen.

Durch die immer größer werdende Akzeptanz von Kamerafallen, steigende Qualität und sinkende Preise kommt es zu einer exponentiell wachsenden Datenmenge. Diesen Daten manuell Herr zu werden stellt eine Herausforderung dar und aufgrund der Verwendung von unveröffentlichten Daten in aktuellen Forschungsvorhaben ist Crowd-Sourcing oft unmöglich. Die anfallenden Bilder zeichnen sich durch einen hohen Anteil von False-Positives und eine große Vielfalt von Arten in verschiedensten Posen, Entferungen und bei wechselnden Witterungsbedingungen aus.

Im Kontext dieser Arbeit beziehen wir uns dabei auf einen Datensatz aus dem niederländischen Nationalpark „Hoge Veluwe“. Die Datenbank umfasst 40 GB Bilder von neun einheimischen Tierspezies, die mithilfe von RECONYX-Kamerafallen gesammelt wurden. Dabei wurden sowohl Farbbilder am Tag als auch Infrarotbilder in der Nacht geschossen.

Um diese komplexe Aufgabe zu automatisieren, stellen wir eine Softwarepipeline vor, mit deren Hilfe es möglich ist alle nacheinander anfallenden Herausforderungen zu lösen. Der erste Schritt besteht in der Ordnung der Daten. Dafür haben wir mit dem *Camera Trap Sequencer* eine Software mit grafischer Benutzeroberfläche implementiert, die es dem Benutzer erlaubt nach Tierarten vorsortierte Datenbanken oder einzelne Ordner von Bildern auf zusammenhängende Sequenzen aufzuteilen.

Der nächste Schritt ist die Lokalisierung von Tierarten in Bildern. Das ermöglicht zum einen das Aussortieren von Bildern, die in Wirklichkeit keine Tiere zeigen, und zum anderen die Identifikation von Bildausschnitten, die für die spätere Klassifizierung relevant sind. Hierfür verwenden wir eine Pipeline mit verschiedenen Vor- und Nachbereitungsschritten, die mithilfe von *Principal Components Analysis* ein Hintergrundbild auf einer Bildsequenz berechnet und somit die Segmentierung von relevanten Bildausschnitten erlaubt. Um auch die Lokalisierung von Tieren auf Einzelbildern zu erlauben, wurde zusätzlich ein PCA-unterstütztes *Sliding-Window-Verfahren* implementiert.

Den Abschluss jeder Auswertung bildet das Klassifizieren von Spezies in zuvor bestimmten *Regions of Interest*. Hierzu stellen wir zwei verschiedene Techniken vor: Die erste ist die Klassifizierung mit Hilfe einer *Support Vector Machine* mit *Radial-Basis-Function*-Kernel auf dem *Histogram-of-oriented-Gradients*-Feature, einem Strukturfeature. Das zweite Verfahren ist *Spatial Pyramid Matching* (SPM) mit *Locality-constrained linear Coding*. Hierbei wird das Eingabebild in immer feinere Teilbilder unterteilt, auf denen dann SIFT oder LBP Features berechnet werden. Diese Features werden mit LLC kodiert, wobei die räumliche Aufteilung erhalten bleibt. Abschließend

werden die so bestimmtem Codes zum Trainieren einer Support Vector Machine mit linearem Kernel benutzt, da sich empirisch erwiesen hat, dass sie gut linear separierbar sind .

Zum Abschluss unserer Ausarbeitung evaluieren wir unsere Verfahren. Betrachtet werden sowohl die Laufzeit der Algorithmen als auch ihre Güte auf den uns zur Verfügung stehenden Daten. Da der ursprüngliche Datensatz zu groß ist, betrachten wir dabei lediglich zwei repräsentative Teilmengen: In der DDD befinden sich Tagbilder von Dachsen und Damhirschen. Die geringe Datenmenge erlaubt schnelle Ergebnisse, ohne grundlegende Probleme, wie beispielsweise unterschiedlich unbalancierte Klassenhäufigkeiten, aus dem Blick zu verlieren. Für die DDD+ haben wir über 2000 Tag- und Nachtbilder von insgesamt sechs verschiedenen Tierarten zusammengestellt. Ziel ist hierbei die Bestimmung der Güte des Verfahrens auf einem komplexen Datensatz.

## **2 Aufteilung auf Sequenzen**

### **2.1 Sortierung mit Hilfe der EXIF-Daten**

-EXIF -EXIF Tool

### **2.2 Camera Trap Sequencer**

-Optionen -Beschreibung des Aufbaus des Datensatzes -Implementierung mit qt5

## 3 Lokalisierung mit Principal Components Analysis (PCA)

Die automatische Lokalisierung von Objekten in digitalen Bildern ist ein wesentlicher Bestandteil vieler Anwendungen. Für das Lokalisierungsproblem in dieser Arbeit bietet sich die Verwendung von der Methoden *Hintergrund-Subtraktion* und *Sliding-window mit PCA* an.

### 3.1 Hintergrundapproximation mit PCA

Um Bewegungen in Bildsequenzen erkennen zu können, wird in der Praxis sehr häufig das Verfahren der *Hintergrund-Subtraktion* angewandt. Dabei handelt es sich um ein klassisches Verfahren aus dem Bereich der Bilderkennung. Das Hintergrundbild kann mithilfe von PCA approximiert werden. Anschließend wird das Vordergrundbild über die Differenz zum Hintergrundbild extrahiert. PCA, oder auch Hauptkomponentenanalyse, ist ein statistisches Verfahren um große Mengen von Datensätzen zu vereinfachen und zu strukturieren, indem die Datenpunkte im  $p$ -dimensionalen Raum  $R^p$  in einen  $q$ -dimensionalen Unterraum  $R^q$  mit ( $q < p$ ) projiziert werden. Diese Transformation muss dabei so gewählt werden, dass möglichst wenig Information verloren geht. Grundsätzlich benutzt PCA die *Niedrigrang Approximation*. Damit kann eine Matrix durch eine andere Matrix im allgemeinen Rang angenähert werden. Sei eine Matrix  $A$  mit  $\text{Rang}(A) = r$  und  $r > k$ :

$$\min_{\text{rang}(A)=k} \|A - B\|_2 \quad (1)$$

Dabei soll die Differenz zwischen  $A$  und  $B$  minimiert werden. Mithilfe der *Singulärwertzerlegung (SVD)* können die Singulärwerte einer Matrix abgelesen werden. Die SVD von Matrix  $A$  ist dann:

$$A = U\Sigma V^T \quad (2)$$

Somit kann ein Hintergrundbild aus einer Sequenz von Bildern wie folgt approximiert werden (Abbildung 1):

- ▷ Berechne Singulärwertzerlegung aller Bildern von Sequenz  $X$ :

$$SVD(X) = C = U\Sigma V^T \quad (3)$$

- ▷ Leite die Matrix  $\Sigma_k$  von  $\Sigma$  her, sodass die Werte  $n - k$  entlang der Diagonale durch 0 ersetzt werden.

- ▷ Dies ergibt die Niedrigrang Approximation vom Matrix  $X$ :

$$SVD(X)_k = C_k = U\Sigma_k V^T \quad \text{mit} \quad \Sigma_k = \text{diag}(\sigma_1, \dots, \sigma_k, 0, \dots, 0) \quad (4)$$

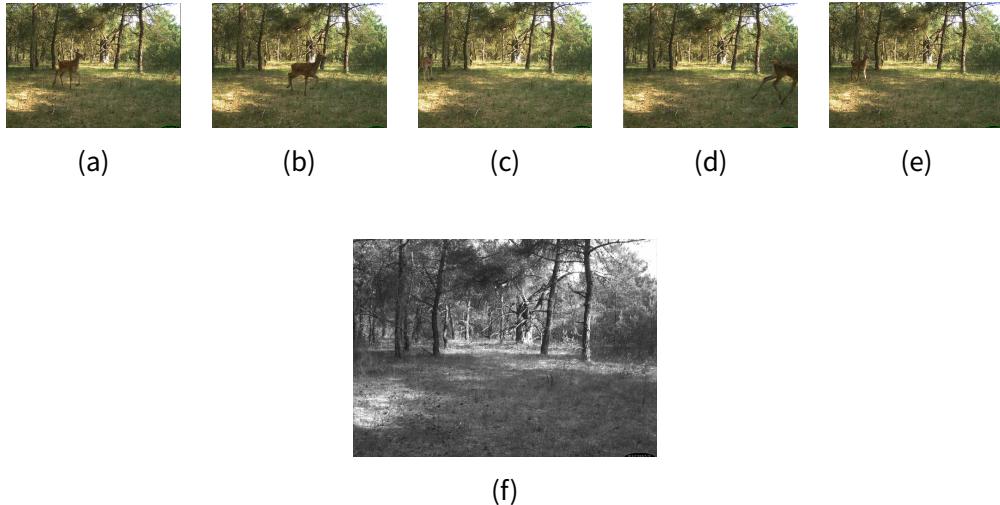


Abbildung 1: (a)-(e) Bilder aus einer Sequenz und (f) das approximierte Hintergrundbild.

Anschließend kann das Vordergrundbild durch die klassische *Hintergrund-Subtraktion* extrahiert werden (Abbildung 2).

$$\begin{array}{ccc} \text{M} & - & \text{L} = \text{S} \end{array}$$

The diagram illustrates the foreground extraction process. It shows two images, M and L, being subtracted to produce the result S. The images are represented as grayscale patches, and the subtraction operation is indicated by a minus sign between them.

Abbildung 2: Das Vordergrundbild ergibt sich durch die Subtraktion des approximierten Hintergrundbildes.

Zum Nachbearbeitung des Vordergrundes gehört eine Vielzahl von Operationen z.B. *morphologische Operationen*, *Thresholding* und *Filterung*. Damit können kleinere Bildstrukturen und Rauschen entfernt, vergrößert, geschlossen oder aufgefüllt werden. Können jedoch diese Operationen zu einer Veränderung der Größe der Vordergrundelemente führen, was zur Lokalisierung des Elements aber keinen Störfaktor ergibt. Durch Kombination der Operationen in einer bestimmten Reihenfolge kann Größenveränderung verhindert und dennoch die Vorteile der Operationen genutzt werden. Durch *Opening* werden zunächst kleine Strukturen bzw. Rauschen, welches zum Hintergrund gehört, entfernt. Danach werden kleine Löcher innerhalb der Vordergrundelemente durch *Closing* geschlossen. In (Abbildung 3) ist eine Kombination dieser Pipeline zur Nachbearbeitung des Vordergrundes benutzt worden.

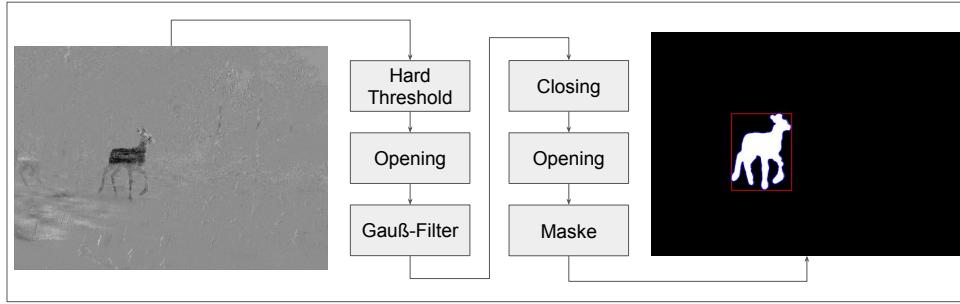


Abbildung 3: Die Pipeline der Nachbearbeitung des Vordergrundbildes. Durch *Opening* und *Closing* werden kleine Bildstrukturen bzw. Rauschen entfernt und kleine Löcher geschlossen werden. Die Gauß-Filterung dient in diesem Fall dazu, die Silhouette des Vordergrundelements grob zu vergrößern.

### 3.2 Sliding-window Lokalisierung mit PCA

Sliding-window ist eine Brute-Force-Suche über das Bild mit fester Fenstergröße, um Objekte zu finden. Für jedes dieser Fenster wird ein Bildklassifikator angewendet, um zu bestimmen, ob das Fenster ein bekanntes Objekt enthält. In diesem Fall wird PCA als Objekt-Klassifikator angewandt.

### 3.3 Objektdetektion mit PCA

Jedes Bild ist ein Punkt in einem hochdimensionalen Raum. Durch das PCA-Verfahren lassen sich die Datenpunkte in einem kleineren dimensionalen Unterraum abbilden. PCA sucht die ersten  $k$ -Hauptkomponenten, welche die Daten mit einer maximalen Varianz beschreiben. Damit wird es eine niederdimensionale Darstellung gefunden, bei der die Klassifizierung leichter wird.

#### 3.3.1 Algorithmus

- ▷ Phase I: Initialisierung
  - ▷ Berechne das Mittelwertbild der Trainingsbilder

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i \quad (5)$$

- ▷ Berechne die zentrierten Daten durch Subtraktion der Trainingsbilder vom Mittelwertbild

$$C = X - \mu \quad (6)$$

- ▷ Berechne die Eigenwerte und Eigenvektoren für die Kovarianzmatrix  $CC^T$

$$\text{SVD}(C) = \mathbf{U}\Sigma\mathbf{V}^T \quad (7)$$

- ▷ Projiziere die Trainingsbilder in den  $r$ -Unterraum

$$\mathbf{Y} = \mathbf{U}_r^T C \quad (8)$$

- ▷ Phase II: Klassifikation

Gegeben ist ein unbekanntes Bild  $M$

- ▷ Projiziere das Bild  $M$  in den  $r$ -Unterraum

$$\mathbf{W} = \mathbf{U}_r^T (M - \mu) \quad (9)$$

- ▷ Finde den nächsten Nachbarn zwischen den projizierten Trainingsbildern  $\mathbf{Y}$  und dem projizierten Bild  $\mathbf{W}$ .

Die Sliding-Windows laufen das Bild mit unterschiedlichen Fenstergrößen durch. Demnach werden Schnittbilder einzelne mit PCA klassifiziert. Dabei wird der nächste Nachbar der projizierten Schnittbilder gefunden und zugeordnet (Abbildung 4).



Abbildung 4: Lokalisierung mit Sliding-windows und PCA.

## **4 Klassifizierung mit Histograms of Gradients und Support Vector Machines**

## **5 Klassifizierung mit Spatial Pyramid Matching**

### **5.1 Grundidee Spatial Pyramid Matching**

### **5.2 Locality-constrained Linear Coding**

### **5.3 Implementierung**

## **6 Evaluierung**

### **6.1 Daten**

### **6.2 Laufzeit Spatial Pyramid Matching**

### **6.3 Klassifizierung Spatial Pyramid Matching**

## **7 Fazit**

## **Eigenständigkeitserklärung**

Hiermit versichern wir, dass die vorliegende Ausarbeitung *Auswertung von Kamerafallenbildern mit Hilfe von Principal Components Analysis, Spatial Pyramid Matching und Support Vector Machines* selbstständig verfasst worden ist, dass keine anderen Quellen und Hilfsmittel als die angegebenen benutzt worden sind und dass die Stellen der Arbeit, die anderen Werken – auch elektronischen Medien – dem Wortlaut oder Sinn nach entnommen wurden, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht worden sind.

---

(Ort, Datum)

---

(Unterschrift)