Computer Vision: Bildmerkmale

D. Schlesinger – TUD/INF/KI/IS

- Block Matching + Clusterung = Templates
- Basisfunktionen + PCA = Haar-Merkmale
- SIFT

Block Matching

Idee – Bilder sind Vektoren, ein Pixel entspricht einer Komponente.
Ein Bildfragment um ein Pixel ist ein Vektor, der die Umgebung „charakterisiert“.

Wie sind zwei Bildfragmente (Bild I_1 an der Stelle $p_1 = (i_1, j_1)$ mit dem Bild I_2 an der Stelle $p_2 = (i_2, j_2)$) zu vergleichen? Zum Beispiel mit dem quadratischen Abstand

$$D\Big(I_1(p_1), I_2(p_2)\Big) = \sum_{p' \in W} \Big(I_1(p_1 + p') - I_2(p_2 + p')\Big)^2$$

Weitere Schritte – gewisse Farbtransformationen zu erlauben.
Zum Beispiel: Die Färbungen dürfen sich um eine additive Konstante unterscheiden (unterschiedliche Helligkeit) – nur der „Rest“ wird beurteilt:

$$D_a\Big(I_1(p_1), I_2(p_2)\Big) = \min_{C_a} \sum_{p' \in W} \Big(I_1(p_1 + p') + C_a - I_2(p_2 + p')\Big)^2$$

Ableitung nach C_a, auf Null setzen, auflösen $\Rightarrow$

$$C_a^* = \frac{1}{|W|} \sum_{p' \in W} \Big(I_2(p_2 + p') - I_1(p_1 + p')\Big)$$

$$D_a\Big(I_1(p_1), I_2(p_2)\Big) = D\Big(I_1(p_1), I_2(p_2)\Big) - C_a^{*\,2} = D\Big(\tilde{I}_1(p_1), \tilde{I}_2(p_2)\Big)$$

Mittelwertfreie Färbungen.

Block Matching

Weitere Invarianten:
Skalierung (Kontrast): $I_1(p_1 + p')$ wird zu $C_m \cdot I_1(p_1 + p') + C_a$
$\Rightarrow$ Korrelationskoeffizient – mittelwertfreie Färbungen, normiert auf ihre Varianzen.

Geometrische Transformationen: $I_1(p_1 + p')$ wird zu $I_1(Tr(p_1 + p'))$
usw.

Zurück zu Bilddeskriptoren:

Idee (Annahme) – Die Menge aller Bildfragmente besteht aus Teilmengen,
jede Teilmenge hat einen Repräsentant – ein idealer Muster,
alle Bildfragmente sind verrauschte Varianten der Repräsentanten.

Der Merkmal eines Bildfragmentes ist der ähnlichste Repräsentant.

Man finde die Repräsentanten anhand einer Lernstichprobe von Bildfragmenten
$\Rightarrow$ Clusterung

Clusterung

Aufgabe: partitioniere eine Menge der Objekte auf sinnvolle Teile – Clusters.
Die Objekte eines Clusters sollen „ähnlich“ sein.

Clustermenge: K
Indexmenge: $I = \{1, 2, \ldots, |I|\}$
Merkmalsvektoren: $x^i,\ i \in I$.
Partitionierung: $C = (I_1, I_2, \ldots, I_{|K|}),\ I_k \cap I_{k'} = \emptyset$ für $k \neq k'$, $\bigcup_k I_k = I$

Jeder Cluster hat einen „Repräsentant“ y^k

Die Aufgabe:

$$\sum_k \sum_{i \in I_k} \|x^i - y^k\|^2 \to \min_{C,y}$$

Alternativ – gesucht wird eine Abbildung $C : I \to K$

$$\sum_i \|x^i - y^{C(i)}\|^2 \to \min_{y,C}$$

$$\sum_i \min_k \|x^i - y^k\|^2 \to \min_y$$

K-Means Algorithmus

Initialisiere Clusterzentren y^k zufällig.
Wiederhole bis sich die Clusterung C ändert:

1) Klassifikation:

$$C(i) = \arg\min_{k'} \|x^i - y^{k'}\|^2 \quad \rightarrow i \in I_k$$

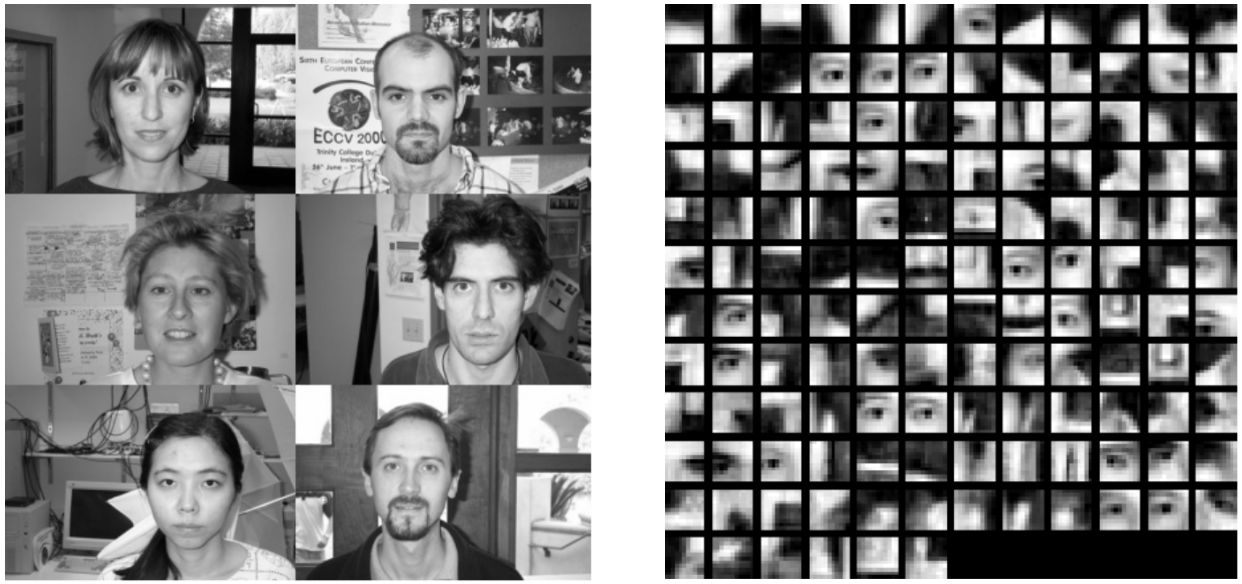
2) Aktualisierung der Zentren:

$$y^k = \arg\min_{y} \sum_{i \in I_k} \|x^i - y\|^2 = \sum_{i \in I_k} x^i$$

- NP-vollständig.
- Bei $|K| \ll |I|$ bleibt kein Cluster frei (bei der global optimalen Lösung).
- K-Means Konvergiert zum lokalen Optimum $\rightarrow$ abhängig von der Initialisierung (Beispiel lokaler Konvergenz auf der Tafel).

Visuelle Wörter

- Finde interessante Punkte in Bildern einer Datenbank
- Betrachte die entsprechenden Bildfragmente als Vektoren
- Clustere sie (Distanzmaß aus Block Matching)

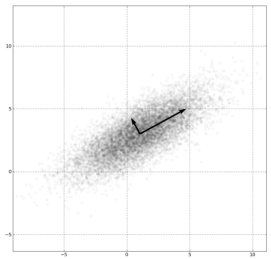


Bilddatenbank $\Rightarrow$ Visuelle Wörter

Jedem Bildausschnitt wird das Wort zugeordnet, zu welchem er am nächsten liegt.
Bildmerkmal – Nummer des Wortes

Hauptkomponentenanalyse (PCA)

Die Vektoren (Bildausschnitte) werden in einem anderen Basis dargestellt.



Annahme: die „Richtungen“ kleiner Varianzen entsprechen dem Rauschen und können vernachlässigt werden.

Die Idee: den Merkmalsraum auf einen linearen Unterraum projizieren so, dass die Varianzen im Unterraum so groß wie möglich sind.

Einfachheit halber – die Daten sind bereits zentriert, der Unterraum ist eindimensional, d.h. durch einen Vektor e mit $\|e\|^2 = 1$ angegeben. Projektion eines x auf e ist $\langle x, e\rangle$.

$$\sum_l \langle x^l, e\rangle^2 \to \max_e \qquad \text{s.t. } \|e\|^2 = 1$$

Lagrange Funktion:

$$\sum_l \langle x^l, e\rangle^2 + \lambda\Big(\|e\|^2 - 1\Big) \to \min_\lambda \max_e$$

Ableitung:

$$\sum_l 2\langle x_l, e\rangle \cdot x_l + 2\lambda e = 2e \sum_l x_l \otimes x_l + 2\lambda e = 0$$

$$e \cdot cov = \lambda e$$

$\to$ e ist Eigenvektor der Kovarianzmatrix, λ ist der entsprechende Eigenwert.

Hauptkomponentenanalyse

Welchen Eigenvektor soll gewählt werden? Die mit einem λ erreichte Varianz ist

$$\sum_l \langle x^l, e\rangle^2 = e \cdot \sum_l x^l \otimes x^l \cdot e = e \cdot cov \cdot e = \|e\|^2 \cdot \lambda = \lambda$$

$\rightarrow$ wähle den Eigenvektor zum größten Eigenwert.

Ähnliche Vorgehensweise: den Merkmalsraum auf einen Unterraum projizieren so, dass die summarische quadratische Abweichung der Datenpunkte von entsprechenden Projektionen so klein wie möglich ist $\rightarrow$ Approximation. Das Ergebnis ist dasselbe.

Zusammenfassend:

1) Berechne die Kovarianzmatrix der Daten $cov = \sum_l x^l \otimes x^l$
2) Suche alle Eigenwerte und Eigenvektoren
3) Ordne sie (fallend) nach Eigenwerten
4) Wähle m Eigenvektoren zu m größten Eigenwerten
5) Die $n \times m$ Projektionsmatrix besteht aus m Spalten, die jeweils die gewählten Eigenvektoren sind.

Welchen Basis zu wählen? Sind die Hauptkomponenten gut?

Funktionsräume

Bilder sind keine Vektoren.
Bilder sind Funktionen $I : D \subset \mathbb{Z}^2 \to C$ (Menge der Pixel in die Menge der Farbwerte).
Bilder sind Funktionen über einen kontinuierlichen Definitionsbereich $I : D \subset \mathbb{R}^2 \to C$
Allerdings kann man eine Funktion auch als ein Vektor verstehen (darstellen).

	Vektor $v = (v_1, v_2 \dots v_n)$	Funktion $f(x),\ x \in \mathbb{R}$
Definitionsbereich	$\{1, 2 \dots n\}$	$\mathbb{R}$
Abbildung	$\{1, 2 \dots n\} \to \mathbb{R}$	$\mathbb{R} \to \mathbb{R}$
Raum	$\mathbb{R}^n$	$\mathbb{R}^\infty$
Skalarprodukt	$\langle u, v \rangle = \sum_i u_i v_i$	$\int f(x) g(x) dx$
Länge	$\sum_i v_i^2 = \langle v, v \rangle$	$\int f(x)^2 dx$

$\Rightarrow$ Bilder sind mehr als Vektoren, Vektoren sind sie aber auch.

Basis im Vektorraum

Die Aufgabe – zerlege einen Vektor $x \in \mathbb{R}^n$ auf seine „Komponenten“ in einem Basis

$$x = \sum_i v_i \cdot \lambda_i,$$

mit den Basisvektoren $v_i \in \mathbb{R}^n$ und den Koeffizienten $\lambda_i \in \mathbb{R}$.
Äquivalent – löse ein lineares Gleichungssystem $x = V \cdot \lambda$ mit $\lambda \in \mathbb{R}^n$

Eigenschaften des Basis:

- Die Basisvektoren sollen den Raum aufspannen, d.h. eine solche Zerlegung existiert für alle x.
- Die Vektoren sind linear unabhängig, d.h. ein Vektor v_i lässt sich nicht als eine lineare Kombination anderer Vektoren v_j darstellen – die Zerlegung eines x ist dann eindeutig.

Spezialfall – orthonormierter Basis:

- alle v_i sind zu einander orthogonal, d.h. $\langle v_i, v_j \rangle = 0$ für $i \neq j$.
- alle v_i haben dieselbe Länge ($= 1$), d.h. $\langle v_i, v_i \rangle = 1$.

Dann gilt: $\lambda_i = \langle x, v_i \rangle$.

Basis im Funktionsraum

Der Funktionsraum ist unendlichdimensional →
→ unendlich viele Basisfunktionen $v_i(x)$, d.h. $v(x, y)$ (y ersetzt i) sowie
→ eine kontinuierliche Funktion $\lambda(y)$.

Die Aufgabe ist, eine gegeben Funktion $f(x)$ auf Basisfunktionen zu zerlegen:

$$f(x) = \int_y v(x, y)\lambda(y)\,dy$$

Orthonormierter Basis heißt:

- Orthogonal:

$$\int_x v(x, y')v(x, y'')\,dx = 0 \quad \text{für alle } y' \neq y'' \ .$$

- Normiert:

$$\int_x v(x, y)v(x, y)\,dx = const \quad \text{für alle } y.$$

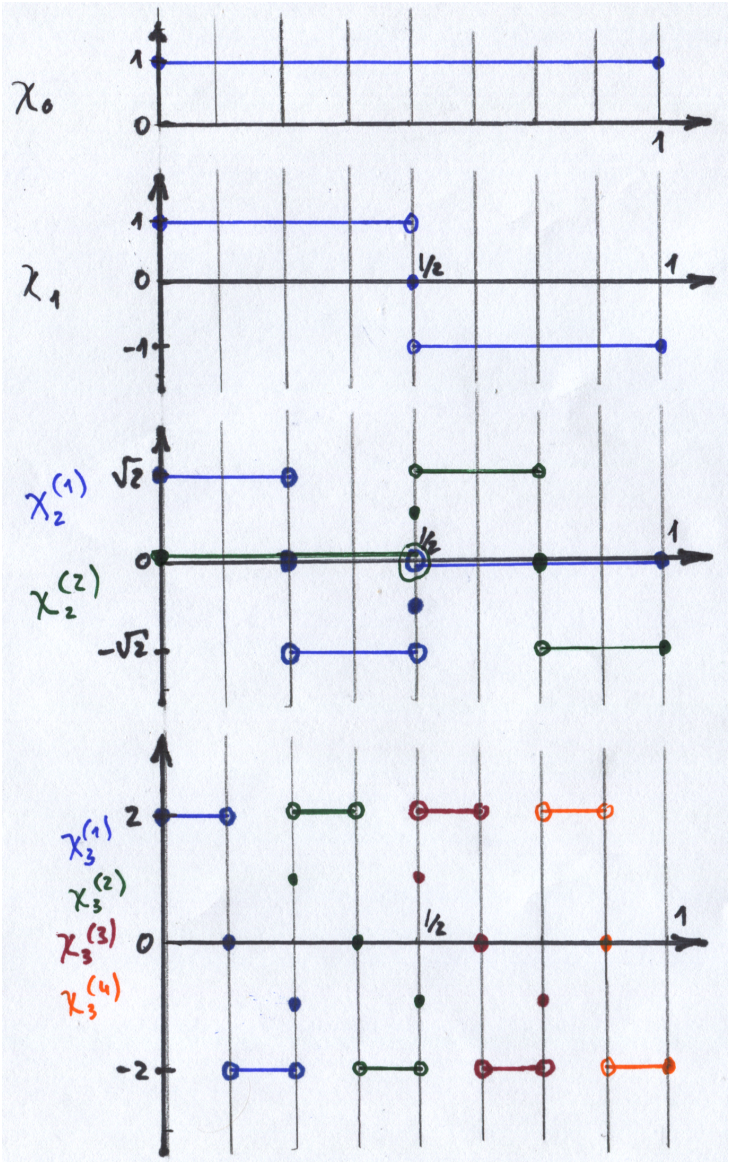
Dann gilt:

$$\lambda(y) = „\langle\rangle“ = \int_x f(x)v(x, y)\,dx$$

Beispiel: Basisfunktionen $\cos(yx)$ und $\sin(yx)$ → Fourier-Transformation (1822)

Basis-Funktionen von Haar (1909)

Eindimensional:



$$\chi_0(s) = 1, 0 \le s \le 1$$

$$\chi_1(s) = \begin{cases} 1 & 0 \le s < 1/2 \\ -1 & 1/2 < s \le 1 \end{cases}$$

$$\chi_2^{(1)}(s) = \begin{cases} \sqrt{2} & 0 \le s < 1/4 \\ -\sqrt{2} & 1/4 < s < 1/2 \\ 0 & \text{sonst} \end{cases}$$

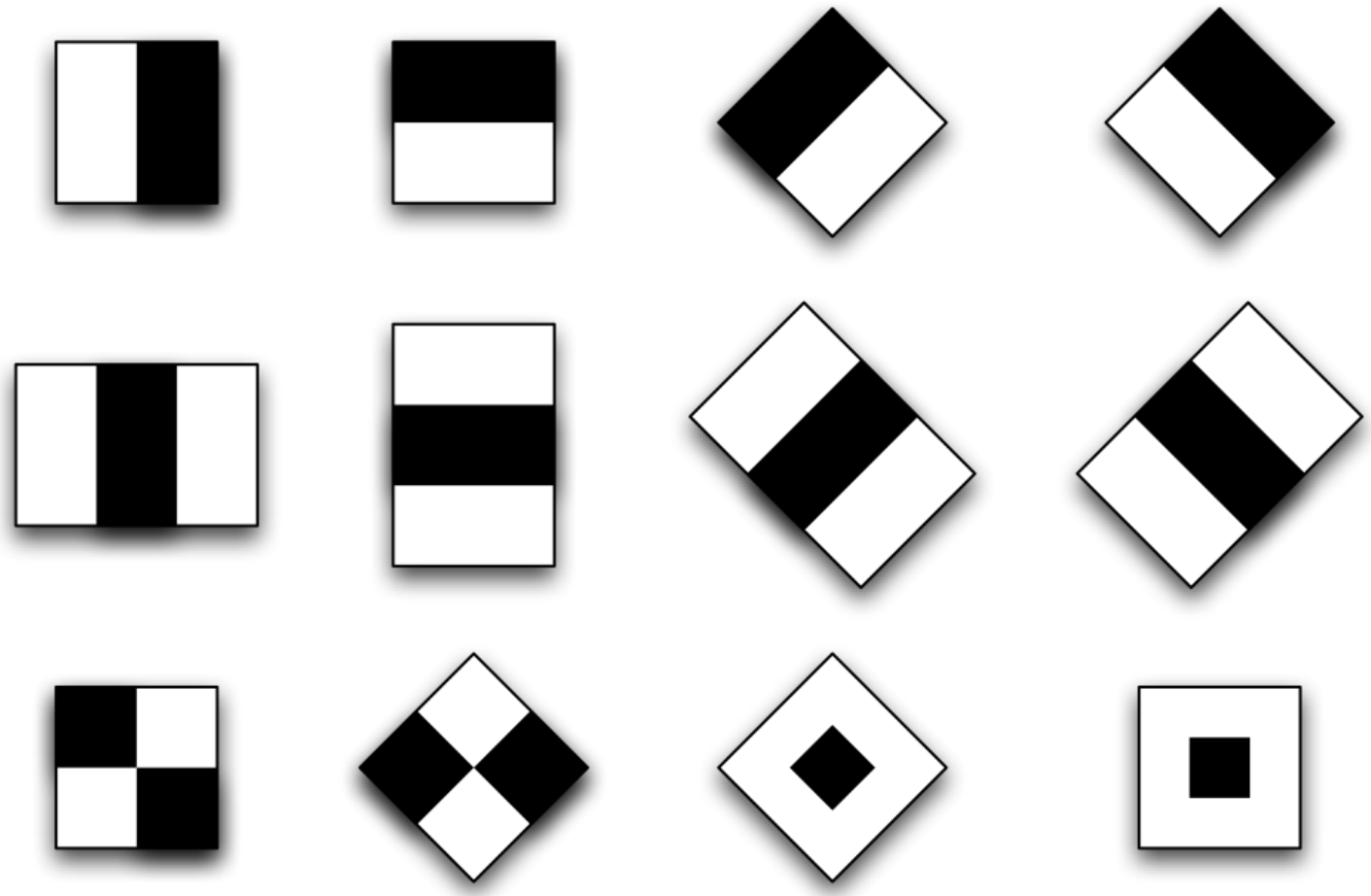
$$\chi_2^{(2)}(s) = \begin{cases} \sqrt{2} & 1/2 < s < 3/4 \\ -\sqrt{2} & 3/4 < s \le 1 \\ 0 & \text{sonst} \end{cases}$$

$$\chi_3^{(1)}, \chi_3^{(2)}, \chi_3^{(3)}, \chi_3^{(4)}$$

usw.

Haar-Bildmerkmale

Filter: ± 1 (Schwarz/Weiß)



Können mithilfe des Integralbildes sehr effizient berechnet werden!!!

1) Interessante Punkte werden gesucht
2) Bildfragmente werden normalisiert (meist affine, seltener projektive Verzerrung)
   z.B. mithilfe der Autokorrelationsfunktion
3) Der Deskriptor ist:

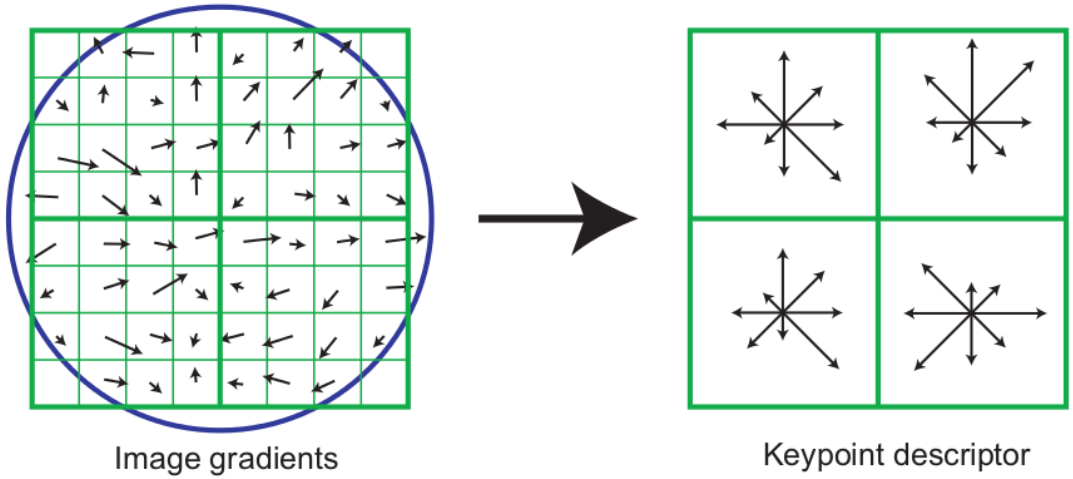


Figure 7: A keypoint descriptor is created by first computing the gradient magnitude and orientation at each image sample point in a region around the keypoint location, as shown on the left. These are weighted by a Gaussian window, indicated by the overlaid circle. These samples are then accumulated into orientation histograms summarizing the contents over 4x4 subregions, as shown on the right, with the length of each arrow corresponding to the sum of the gradient magnitudes near that direction within the region. This figure shows a 2x2 descriptor array computed from an 8x8 set of samples, whereas the experiments in this paper use 4x4 descriptors computed from a 16x16 sample array.

Figure 12: The training images for two objects are shown on the left. These can be recognized in a cluttered image with extensive occlusion, shown in the middle. The results of recognition are shown on the right. A parallelogram is drawn around each recognized object showing the boundaries of the original training image under the affine transformation solved for during recognition. Smaller squares indicate the keypoints that were used for recognition.

Es ist bisher nicht klar, warum der SIFT in der Praxis so gut funktioniert.

Literatur

– Alfred Haar in Göttingen: Zur Theorie der orthogonalen Funktionensysteme (Erste Mitteilung)

  On the Theory of Orthogonal Function Systems (First communication)

– David G. Lowe: Distinctive Image Features from Scale-Invariant Keypoints