Computer Vision: SVM-Anwendungsbeispiele, Generalisierbarkeit

D. Schlesinger – TUD/INF/KI/IS

- Visual Categorization with Bags of Keypoints
- Recognizing Human Actions: A Local SVM Approach
- Shape Matching and Object Recognition Using Shape Contexts
- SkyFinder: Attribute-based Sky Image Search
- Generalisierbarkeit, VC-Dimension

Visual Categorization with Bags of Keypoints

Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski, Cédric Bray

... bag of keypoints method is based on vector quantization of affine invariant descriptors of image patches

image patches: Bildfragmente (spielen in dieser Arbeit kaum eine Rolle)
affine invariant descriptors: Harris+Laplace Detektor, Normalisierung, SIFT

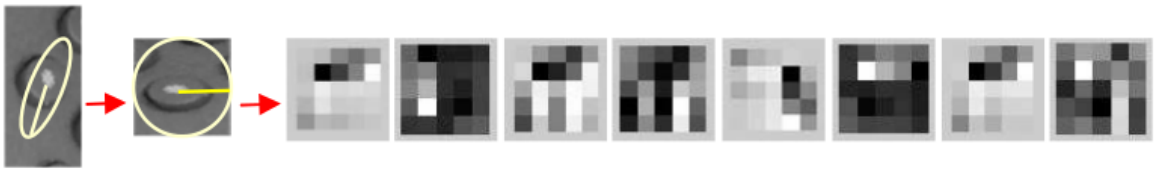


Fig. 1. (*From left to right*) A Harris affine region; the normalized region; and the 8 maps of gradient magnitude constituting the SIFT descriptor.

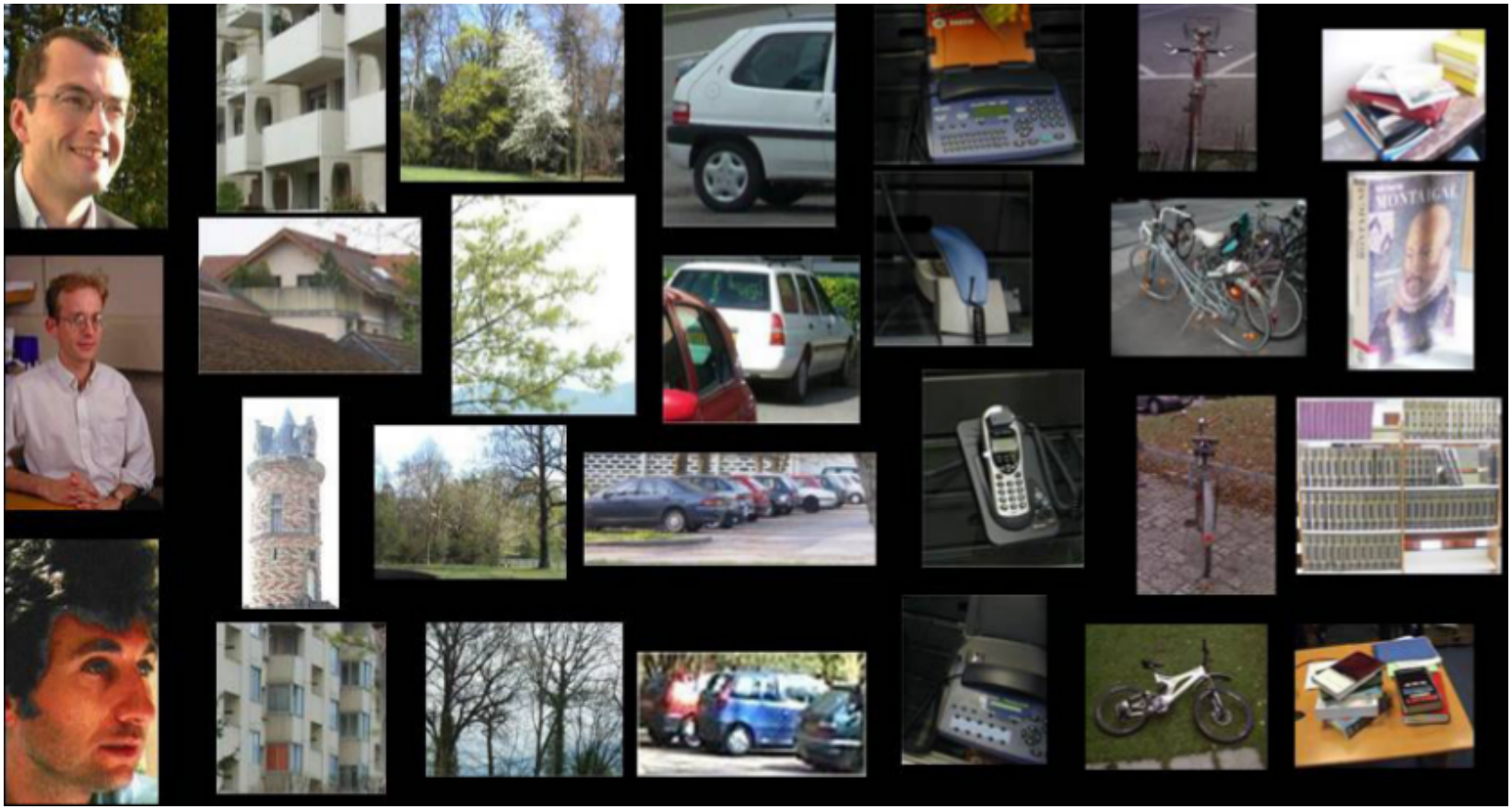
vector quantization: Clusterung mit K-Means für $\|\cdot\|^2$, mit zufällige Initialisierungen
bag of keypoints: Histogramm des Vorkommens der Clusternummern $\rightarrow$ Bildmerkmal

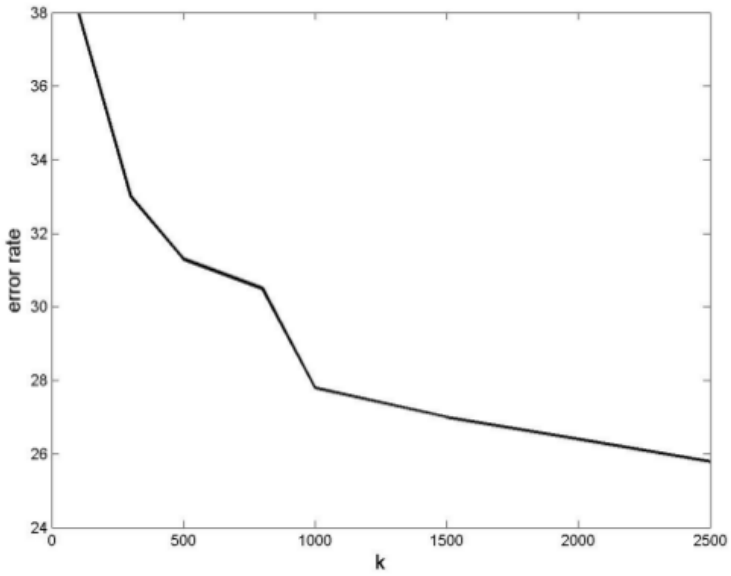
Klassifikation: „Naïve Bayes“ und SVM (one-against-all)

??? *We compared linear, quadratic and cubic SVM's and found that linear method gave the best performance (except in the case of cars where a quadratic SVM gave better results).*

Visual Categorization with Bags of Keypoints

Bilddatenbank: 1776 Bilder, 7 Klassen: faces, buildings, trees, cars, phones, bikes, books





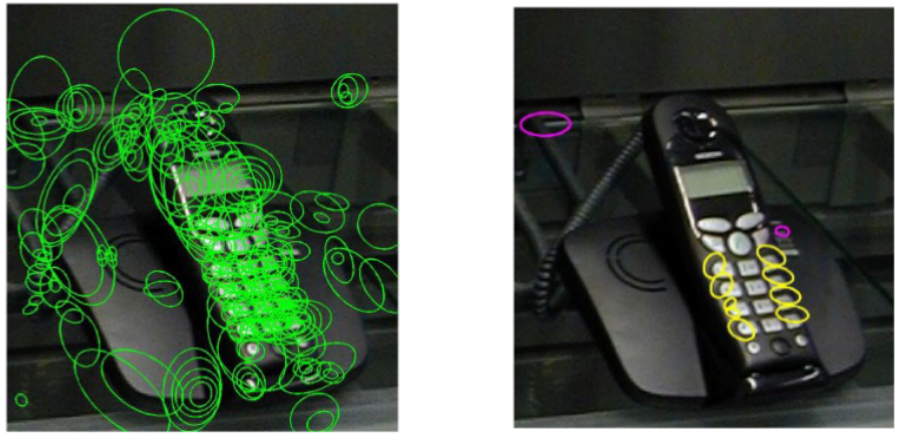


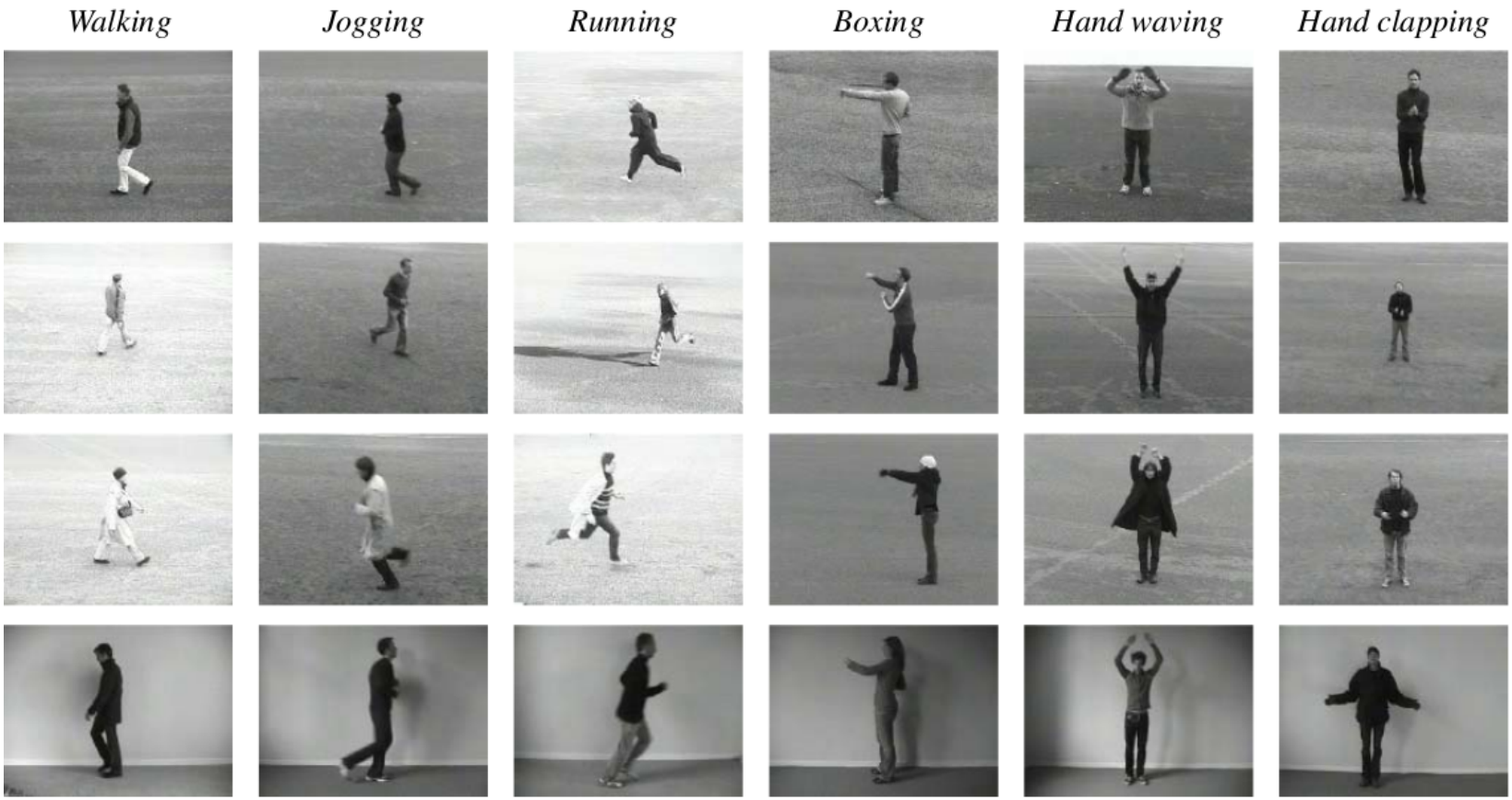
Fig. 4. *Left* all patches detected for this image. *Right* patches from two selected clusters occurring in this image (yellow and magenta ellipses).

Confusion Matrix:

True classes →	*faces*	*buildings*	*trees*	*cars*	*phones*	*bikes*	*books*
faces	**98**	14	10	10	34	0	13
buildings	1	**63**	3	0	3	1	6
trees	1	10	**81**	1	0	6	0
cars	0	1	1	**85**	5	0	5
phones	0	5	4	3	**55**	2	3
bikes	0	4	1	0	1	**91**	0
books	0	3	0	1	2	0	**73**
Mean ranks	1.04	1.77	1.28	1.30	1.83	1.09	1.39

Recognizing Human Actions: A Local SVM Approach

Christian Schüldt, Ivan Laptev, Barbara Caputo



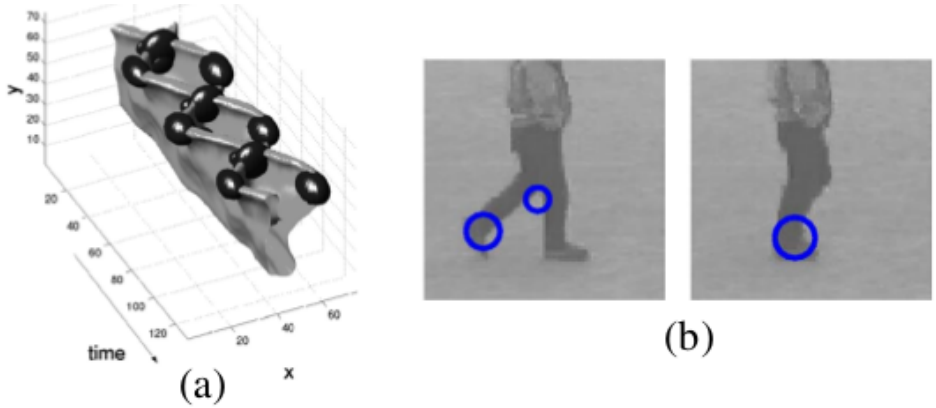
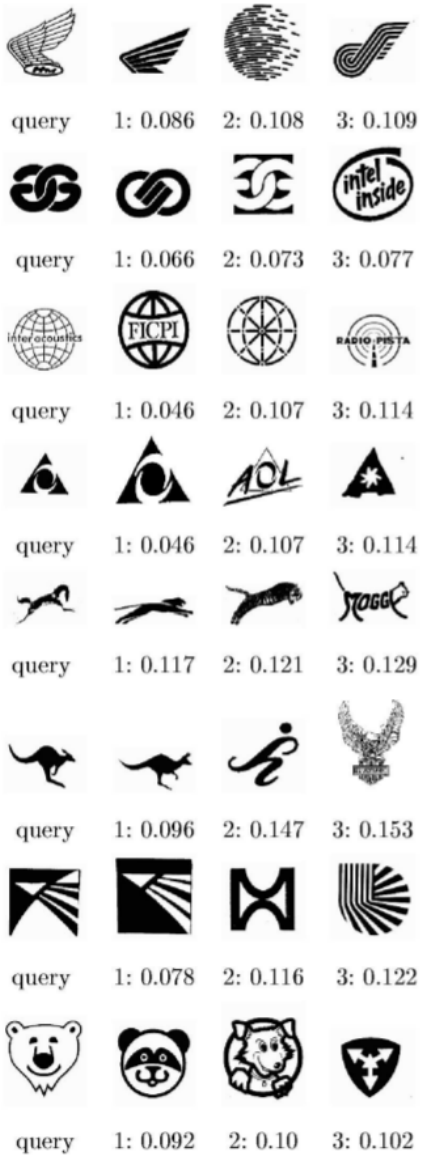


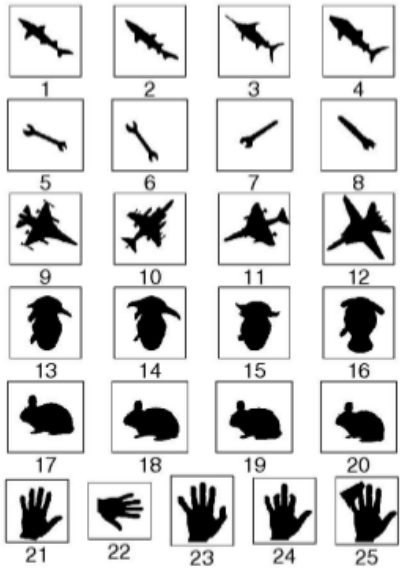
Figure 1. Local space-time features detected for a walking pattern: (a) 3-D plot of a spatio-temporal leg motion (up side down) and corresponding features (in black); (b) Features overlaid on selected frames of a sequence.

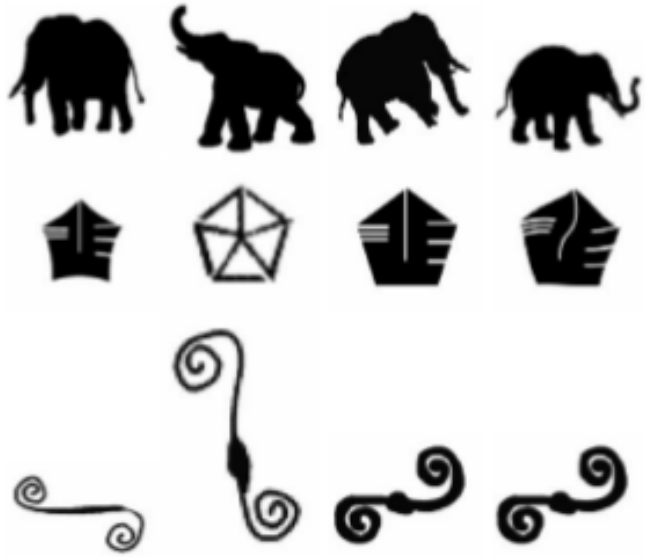
- (Fast) Harris Detektor in 3D (Raum + Zeit + Scale space)
- Merkmale – „spatio-temporal jets“ $l = (L_x, L_y, L_t, L_{xx}, \ldots, L_{tttt})$
- Alternative – K-Means Clusterung + Histogramme
- SVM mit einem speziell dafür entwickelten Kernel (χ^2 für Histogramme)

Shape Matching and Object Recognition Using Shape Contexts

Serge Belongie, Jitendra Malik, Jan Puzicha





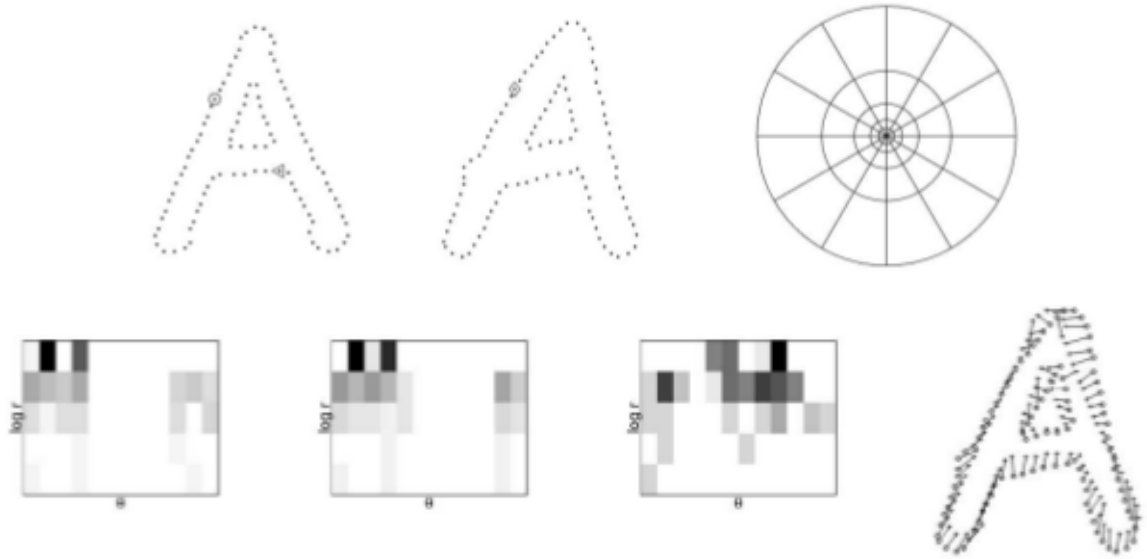


Idee: Abstandsmaß zwischen Shapes (Punktwolken):

- Merkmal für jeden Shape-Punkt
- Zwei Shapes bestmöglich auf einander „matchen“
- Der Abstand ist die Qualität des Matchings

Erkennung anhand des so definierten Abstandsmaßes

Shape Matching and Object Recognition Using Shape Contexts



Shape Context für einen Punkt p ist das Histogramm (in Polarkoordinaten) der Verschiebungsvektoren von p zu allen anderen Shape-Punkten.

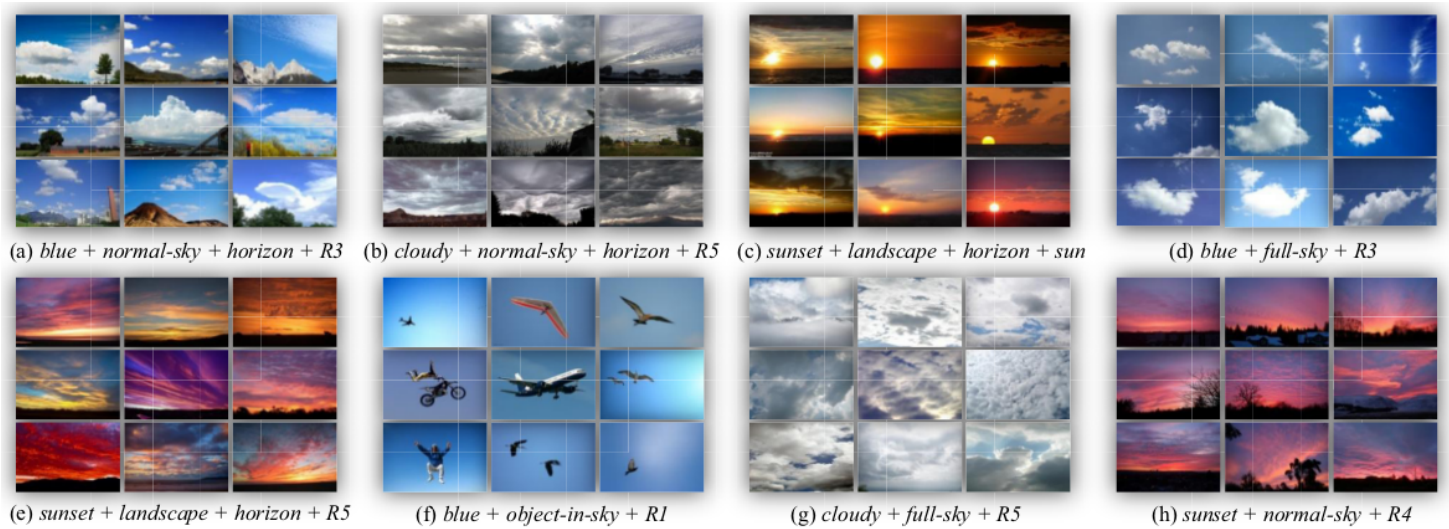
Matching ist die Suche nach besten Korrespondenzen (eine zusätzliche geometrische Transformation wird dabei erlaubt). So entstehen die „Abstände“ für alle Paare von Shapes.

Die Menge aller Shapes wird mit K-*medioids* Algorithmus geclustert. Jeder Cluster wird durch einen Shape repräsentiert, *der in der Lernstichprobe vorhanden ist.*

Erkennung – Nächste Nachbar. Abstände zu allen Repräsentanten werden berechnet, die Klasse des besten wird übernommen.

SkyFinder: Attribute-based Sky Image Search

Litian Tao, Lu Yuan, Jian Sun



(a) *blue + normal-sky + horizon + R3* (b) *cloudy + normal-sky + horizon + R5* (c) *sunset + landscape + horizon + sun* (d) *blue + full-sky + R3*

(e) *sunset + landscape + horizon + R5* (f) *blue + object-in-sky + R1* (g) *cloudy + full-sky + R5* (h) *sunset + normal-sky + R4*

Content Based Image Retrival – CBIR

Der Nutzer gibt die gewünschte Attribute des gesuchten Bildes an
(zum Beispiel „Einen Sonnenuntergang mit Wolken“)
Die Menge der Bilder wird zurückgeliefert, die diese Attribute haben

Die Aufgabe – berechne (schnell), ob ein gegebenes Bild die geforderte Eigenschaft besitzt.

Lösungsweg: SVM für jeden Attribut (binäre Klassifikation).

Besonderheiten – neben der SIFT+HSV spezielle Merkmale (z.B. Lage des Horizonts), eine sehr große Lernstichprobe (Internet)

Generalisierbarkeit

Die Frage: wie genau ist das Lernen?

Es gibt eine Wahrscheinlichkeitsverteilung $p(x, k)$ für Klassen k und Beobachtungen x

Zum Lernen hat man aber nur eine endliche Lernstichprobe
$L = \Big((x_1, k_1), (x_2, k_2) \dots (x_n, k_n)\Big)$

Gegeben sei eine Menge der Entscheidungsstrategien, man wähle eine davon.

Selbst wenn das bestmögliche mit der Lernstichprobe gemacht wird, gibt es immer noch Fehler bei der Erkennung:

1. Fehler auf der Lernstichprobe – das empirische Risiko R_e
2. Fehler auf Gesamtmenge aller x – das Bayessche Risiko R_b

Interessant ist der zweite, berechnen kann man nur den ersten.

Wie groß muss L sein, damit sich R_e und R_b nicht wesentlich unterscheiden?

Wie schnell (und ob überhaupt) konvergiert R_e zu R_b mit der wachsenden L?

Wie hängt die Konvergenz von der Menge der Entscheidungsstrategien ab?

Vapnik–Chervonenkis Dimension

Obere Schranke für die Differenz der Fehler:

$$P\left\{|R_b - R_e| < \sqrt{\frac{h\Big(\log(2N/h)+1\Big) - \log(\delta/4)}{N}}\right\} > 1-\delta$$

mit **VC-Dimension** h
– ein Maß für die „Mächtigkeit“ der Menge der Entscheidungsstrategien.

Klassifizierte Lernstichprobe: $L = \Big((x_1, k_1), (x_2, k_2) \dots (x_m, k_m)\Big)$
Tupel von Punkten: $(x_1, x_2 \dots x_n)$ – nicht klassifizierte Lernstichprobe
Menge der Entscheidungsstrategien: $\mathcal{E}$

Ein Tupel von Punkten ist **beliebig klassifizierbar**, wenn für eine beliebige Lernstichprobe mit den Punkten ein Klassifikator aus $\mathcal{E}$ existiert, die diese Klassifikation wiedergibt.

Die VC-Dimension einer Menge der Strategien ist
die **kleinste** Zahl n so, dass **kein** $(n+1)$-Tupel von Punkten **beliebig klassifizierbar** ist.

Beispiel: lineare Klassifikatoren im $\mathbb{R}^n$, VC$=n+1$.