# Computer Vision <br> 4. Interest Points 

## Lars Schmidt-Thieme

Information Systems and Machine Learning Lab (ISMLL) Institute for Computer Science<br>University of Hildesheim, Germany

## Outline

1. Smoothing, Image Derivatives, Convolutions
2. Edges, Corners, and Interest Points
3. Image Patch Descriptors
4. Interest Point Matching
5. A Simple Application: Image Stitching

## Outline

# 1. Smoothing, Image Derivatives, Convolutions 

2. Edges, Corners, and Interest Points
3. Image Patch Descriptors
4. Interest Point Matching
5. A Simple Application: Image Stitching

## Smoothing / Blurring / Averaging

- Smoothing: Replace each pixel by the weighted average of its surrounding patch:

$$
\begin{aligned}
I_{\text {smooth }}(x, y ; w) & :=\sum_{\Delta x, \Delta y} w(-\Delta x,-\Delta y) l(x+\Delta x, y+\Delta y) \\
& =\sum_{x^{\prime}, y^{\prime}} w\left(x-x^{\prime}, y-y^{\prime}\right) l\left(x^{\prime}, y^{\prime}\right)
\end{aligned}
$$

- padding with 0 at the image boundaries.
- example: box kernel

$$
w_{-2: 2,-2: 2}(\Delta x, \Delta y):=\frac{1}{25}\left(\begin{array}{ccccc}
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1
\end{array}\right)
$$

- Gaussian smoothing: smoothing with a Gaussian kernel.


## Gaussian Kernels

- Precomputed weights: (clipped) Gaussian density values

$$
\begin{aligned}
& \tilde{w}(\Delta x, \Delta y):= \begin{cases}\mathcal{N}\left(\sqrt{\Delta x^{2}+\Delta y^{2}} ; 0, \sigma^{2}\right), & \text { if }|\Delta x| \leq K,|\Delta y| \leq K \\
0, & \text { else }\end{cases} \\
& w(\Delta x, \Delta y):=\frac{\tilde{w}(\Delta x, \Delta y)}{\sum_{\Delta x^{\prime}, \Delta y^{\prime}} \tilde{w}(\Delta x, \Delta y)}
\end{aligned}
$$

- clipped: small support, window size $K$.
- example $\left(K=2, \sigma^{2}=1\right)$ :

$$
w_{-2: 2,-2: 2}:=\left(\begin{array}{ccccc}
0.003 & 0.013 & 0.022 & 0.013 & 0.003 \\
0.013 & 0.060 & 0.098 & 0.060 & 0.013 \\
0.022 & 0.098 & 0.162 & 0.098 & 0.022 \\
0.013 & 0.060 & 0.098 & 0.060 & 0.013 \\
0.003 & 0.013 & 0.022 & 0.013 & 0.003
\end{array}\right)
$$

Note: $\mathcal{N}\left(x ; \mu, \sigma^{2}\right):=\frac{1}{\sqrt{2 \pi \sigma^{2}}} e^{-\frac{(x-\mu)^{2}}{2 \sigma^{2}}}$.
Lars Schmidt-Thieme, Informa and Machine Learning Lab (ISMLL), University of Hildesheim, Germany

## Blurring / Example

original:

blurred by $G(K=5, \sigma=1)$ :


## Blurring / Example

original:

blurred by $G(K=5, \sigma=10)$ :


## Blurring / Example

original:

blurred by $G(K=50, \sigma=1)$ :


## Blurring / Example

original:

blurred by $G(K=50, \sigma=10)$ :


## Image Derivatives

- Image Derivative: How does the intensity values change in x or y direction?

$$
\begin{aligned}
& I_{X}(x, y):=I(x, y)-I(x-1, y) \\
& I_{Y}(x, y):=I(x, y)-I(x, y-1)
\end{aligned}
$$

or symmetric

$$
\begin{aligned}
& I_{X}(x, y):=2 I(x, y)-I(x-1, y)-I(x+1, y) \\
& I_{Y}(x, y):=2 I(x, y)-I(x, y-1)-(x, y-2)
\end{aligned}
$$

## Image Derivatives / Example

original (grayscale):

derivative in $x$-direction:


## Image Derivatives / Example

original (grayscale):

derivative in $y$-direction:


## Convolutions

- Smoothing, Image Derivatives and further operations such as filtering can be represented by a
- convolution: an image where each pixel $(x, y)$ represents the weighted sum around $(x, y)$ in image $I$ weighted with $w$ :
- Examples:

$$
(w * I)(x, y):=\sum_{x^{\prime}, y^{\prime}} w\left(x-x^{\prime}, y-y^{\prime}\right) I\left(x^{\prime}, y^{\prime}\right)
$$

$$
\begin{array}{rlrl}
I_{\text {smooth }} & & =w * I \\
I_{X}(x, y):=I(x, y)-I(x-1, y) & & \left(\begin{array}{cc}
1 & -1
\end{array}\right) * I \\
I_{Y}(x, y):=I(x, y)-I(x, y-1) & & \binom{1}{-1} * I \\
\text { or } I_{X}(x, y):=2 I(x, y)-I(x-1, y)-I(x+1, y) & =\left(\begin{array}{cc}
-1 & 2
\end{array}-1\right) * \\
I_{Y}(x, y):=2 I(x, y)-I(x, y-1)-(x, y+1) & & =\left(\begin{array}{c}
-1 \\
2 \\
-1
\end{array}\right) * I
\end{array}
$$

## Convolutions / Associativity

- Convolutions are associative:

$$
I *(J * K)=(I * J) * K
$$

- Example:

First smooth an image with Gaussian w from slide 2, then compute its $x$-derivative with $\left(\begin{array}{ccc}-1 & 2 & -1\end{array}\right)$ :
$\rightsquigarrow$ just convolve with $\left(\begin{array}{ccc}-1 & 2 & -1\end{array}\right) * w$

$$
\left(\begin{array}{lll}
-1 & 2 & -1
\end{array}\right) * w=\left(\begin{array}{ccccc}
-0.007 & 0.002 & 0.017 & 0.002 & -0.007 \\
-0.033 & 0.008 & 0.077 & 0.008 & -0.033 \\
-0.054 & 0.077 & 0.128 & 0.077 & -0.054 \\
-0.033 & 0.008 & 0.077 & 0.008 & -0.033 \\
-0.007 & 0.002 & 0.017 & 0.002 & -0.007
\end{array}\right)
$$

## Outline

## 1. Smoothing, Image Derivatives, Convolutions

## 2. Edges, Corners, and Interest Points

3. Image Patch Descriptors
4. Interest Point Matching
5. A Simple Application: Image Stitching

## Edges, Corners, and Interest Points

- good candidates for points that are easy to recognize and match in two images are
- points on edges
- corners
i.e., points with sudden intensity changes.
- two stage approach: given an image $I \in \mathbb{R}^{N \times M}$,

1. compute an interestingness measure $i \in \mathbb{R}^{N \times M}$ for points,
2. select a useful set of points $p_{1}, \ldots, p_{K} \in[N] \times[M]$

- with high interestingness measure
- not too close to each other.
- many names: corners, interest points, keypoints, salient points, ...

Note: $[N]:=\{1, \ldots, N\}$.

## Gradient Magnitude (Canny Edge Detector)

- Simply use the magnitude of the gradient as interestingness measure:

$$
i(x, y)=\sqrt{\left(D_{X} * I\right)(x, y)^{2}+\left(D_{Y} * I\right)(x, y)^{2}}
$$

- $D_{X}, D_{Y}$ : differentiation kernels, e.g.,

$$
D_{X}:=\left(\begin{array}{lll}
-1 & 2 & -1
\end{array}\right), \quad D_{Y}:=\left(\begin{array}{c}
-1 \\
2 \\
-1
\end{array}\right)
$$

## Gradient Magnitude / Example

original (grayscale):

gradient magnitude:


## Gradient Magnitude / Example

original (grayscale):

overlay with 500 interest points:


## Laplacian of Gaussian and Difference of Gaussian

Further simple interestingness measures:

- Laplacian of Gaussian (LoG):

$$
i(x, y)=\left(\left(\left(D_{X} * D_{X}+D_{Y} * D_{Y}\right) * G\right) * I\right)(x, y)
$$

- uses second order information
- Difference of two Gaussians (DoG):

$$
i(x, y)=\left(\left(G_{\sigma_{1}}-G_{\sigma_{2}}\right) * I\right)(x, y), \quad \sigma_{1} \neq \sigma_{2}
$$

- uses variations at different scales
- often interpreted as limit of Laplacian of Gaussians

$$
\left(\left(D_{X} * D_{X}+D_{Y} * D_{Y}\right) * G_{\sigma}\right) * I \approx \frac{\sigma}{\Delta \sigma}\left(\left(G_{\sigma+\Delta \sigma}-G_{\sigma-\Delta \sigma}\right) * I\right)
$$

## Harris Corner Detector

- Represent a corner by its patch surrounding it, represent such a patch by a weight function

$$
w:[N] \times[M] \rightarrow \mathbb{R},
$$

i.e.,

$$
w(x, y):= \begin{cases}1, & \text { if }\left|x-x_{0}\right|<3 \text { and }\left|y-y_{0}\right|<3 \\ 0, & \text { else }\end{cases}
$$

for a rectangular patch of size 5 centered around $\left(x_{0}, y_{0}\right)$.

- A point is easy to identify, if its minimum in the autocorrelation surface is pronounced:

$$
E(\Delta x, \Delta y ; w):=\sum_{x, y} w(x, y)(I(x+\Delta x, y+\Delta y)-I(x, y))^{2}
$$

## Harris Corner Detector / Autocorrelation Surface


(a)


Note: left to right: flower bed, roof edge, cloud.
[Sze11, p. 187]

## Harris Corner Detector

$$
E(\Delta x, \Delta y ; w):=\sum_{x, y} w(x, y)(I(x+\Delta x, y+\Delta y)-I(x, y))^{2}
$$

with Hessian at minimum:

$$
\begin{aligned}
H(0,0 ; w) & \approx 2 \sum_{x, y} w(x, y) \nabla I_{(x, y)} \nabla I_{(x, y)}^{T}, \quad \text { for } \frac{\partial^{2} I}{\partial^{2}(x, y)}:=0 \\
& =2 w *\left(\begin{array}{cc}
\left(I_{X}\right)^{2} & I_{X} I_{Y} \\
I_{X} I_{Y} & \left(I_{Y}\right)^{2}
\end{array}\right), \\
I_{X}(x, y) & :=I(x+1, y)-I(x, y) \approx \frac{\partial I}{\partial x}(x, y) \\
I_{Y}(x, y) & :=I(x, y+1)-I(x, y) \approx \frac{\partial I}{\partial y}(x, y)
\end{aligned}
$$

Note: $I * J(x, y):=\sum_{x^{\prime}, y^{\prime}} I\left(x-x^{\prime}, y-y^{\prime}\right) J\left(x^{\prime}, y^{\prime}\right)$ convolution of two images.

## Harris Corner Detector

use SVD to assess steepness

$$
H=U\left(\begin{array}{cc}
\sigma_{1} & 0 \\
0 & \sigma_{2}
\end{array}\right) U^{T}, \quad \sigma_{1} \geq \sigma_{2} \geq 0, U U^{T}=I
$$

and define interestingness measure:
$i_{S h i-T o m a s i}(x, y):=\sigma_{2}$

$$
\begin{array}{ll}
i_{\text {Harris }}(x, y):=\sigma_{1} \sigma_{2}-\alpha\left(\sigma_{1}+\sigma_{2}\right)^{2}=\operatorname{det} H-\alpha \operatorname{trace}(H)^{2}, & \alpha:=0.06 \\
i_{\text {Triggs }}(x, y):=\sigma_{2}-\alpha \sigma_{1}, & \alpha:=0.05
\end{array}
$$

$i_{\text {Brown }}(x, y):=\sigma_{1} \sigma_{2} /\left(\sigma_{1}+\sigma_{2}\right)=\operatorname{det} H / \operatorname{trace}(H)$

- the larger $\sigma_{1: 2}$, the steeper the autocorrelation surface $E$.
- Harris and Brown avoid computing $\sigma_{1}, \sigma_{2}$ explicitly (which requires computing a square root).


## Harris Corner Detector / Algorithm

1: procedure interestpoints-harris $\left(I \in \mathbb{R}^{N \times M} ; w \in \mathbb{R}^{-K: K \times-L: L}, \alpha \in \mathbb{R}\right)$
2: $\quad I_{X}:=D_{X} * I$
3: $\quad I_{Y}:=D_{Y} * I$
4: $\quad I_{X}^{2}:=I_{X} \cdot I_{X}$
5: $\quad I_{Y}^{2}:=I_{Y} \cdot I_{Y}$
6: $\quad I_{X} I_{Y}:=I_{X} \cdot I_{Y}$
7: $\quad A:=w * I_{X}^{2}$ $\triangleright$ compute $H(x, y)=\left(\begin{array}{ll}A(x, y) & C(x, y) \\ C(x, y) & B(x, y)\end{array}\right)$
8: $\quad B:=w * I_{Y}^{2}$
9: $\quad C:=w * I_{X} I_{Y}$
10: $\quad i:=A \cdot B-C \cdot C-\alpha(A+B) \cdot(A+B)$
11: return $i$

- $D_{X}, D_{Y}$ : differentiation kernels, e.g.,

$$
D_{X}:=\left(\begin{array}{lll}
-1 & 2 & -1
\end{array}\right), D_{Y}:=\left(\begin{array}{c}
-1 \\
2 \\
-1
\end{array}\right)
$$

Note: • denotes the element/pixelwise product.

## Harris Corner Detector / Example


(a)

(b)

(c)
a) original, b) Harris corners, c) DoG interest points
[Sze11, p. 213]

## Interest Points at Different Scales (SIFT Detector)

- Interest points also can be identified at different scales in parallel:

$$
i(p, s):=\left(G_{\sigma_{s+1}} * I-G_{\sigma_{s}} * I\right), \quad s \in[S]
$$

where

$$
\sigma_{1}>\sigma_{2}>\cdots>\sigma_{S}
$$

where $S \in \mathbb{N}$ is the number of scale levels

- Often scale levels are grouped by octaves:
- each octave is represented by a downsampling by a factor 2
- scales within an octave are $\sigma_{s}:=2^{s / S_{o}} \sigma$ (with $S_{o}$ the number of scale levels within an ocatve)


## Interest Points at Different Scales (SIFT Detector)


(a)
[Sze11, p. 216]

## Non-Maximum Suppression

- Often neighbors of interest points have similar high interestingness, yielding redundant close-by interest points.
- Keep only interest points that are local maxima in their neighborhood:

$$
i^{\prime}(p):=\left\{\begin{array}{ll}
i(p), & \text { if } i(p)>i\left(p^{\prime}\right) \forall p^{\prime} \in N(p) \\
0, & \text { else }
\end{array} \quad \quad p \in[N] \times[M]\right.
$$

with neighborhood

$$
\begin{array}{ll}
N_{K}(p):=\left\{p^{\prime} \in[N] \times[M]| | p_{x}-p_{x}^{\prime}\left|\leq K,\left|p_{y}-p_{y}^{\prime}\right| \leq K, p^{\prime} \neq p\right\}\right. & \\
N_{K}(p):=\left\{p^{\prime} \in[N] \times[M] \mid\left\|p-p^{\prime}\right\| \leq K, p^{\prime} \neq p\right\} &
\end{array}
$$

## Non-Maximum Suppression / Example



Note: $\mathrm{ANMS}=$ adaptive non-maximum suppression; see the book for detalifzze11, p. 214]

## Non-Maximum Suppression / At Different Scale

- Non-Maximum Suppression also can be extended to work on interest points at different scale:

$$
\begin{gathered}
N_{K}(p, s):=\left\{\left(p^{\prime}, s^{\prime}\right) \in[N] \times[M] \times[S]| | p_{x}-p_{x}^{\prime}\left|\leq K,\left|p_{y}-p_{y}^{\prime}\right| \leq K,\right.\right. \\
\left.\left|s-s^{\prime}\right| \leq 1, p^{\prime} \neq p\right\}
\end{gathered}
$$


[Sze11, p. 216]

## SIFT Interest Points

SIFT refines interest points by further steps:

- non-maximum suppression at different scale
- localization of interest points at sub-pixel granularity


## Outline

## 1. Smoothing, Image Derivatives, Convolutions

2. Edges, Corners, and Interest Points
3. Image Patch Descriptors
4. Interest Point Matching
5. A Simple Application: Image Stitching

## Image Patch Descriptors

- Which properties from a patch to extract?
- grayscale intensities, color intensities, gradient directions
- Which patches to extract?
- orientation of the patch w.r.t. the image frame
- offset of the patch w.r.t. the interest point (cells)


## Histograms

- the most simple patch:
- a square centered on the interest point


## Histograms

- the most simple patch:
- a square centered on the interest point
- properties:
- most simple: grayscale intensities of the pixels
- how to represent?
- as a matrix or a vector


## Histograms

- the most simple patch:
- a square centered on the interest point
- properties:
- most simple: grayscale intensities of the pixels
- how to represent?
- as a matrix or a vector
- is affected by rotations


## Histograms

- the most simple patch:
- a square centered on the interest point
- properties:
- most simple: grayscale intensities of the pixels
- how to represent?
- as a matrix or a vector
- is affected by rotations
- by some scalar properties (mean, standard deviation)


## Histograms

- the most simple patch:
- a square centered on the interest point
- properties:
- most simple: grayscale intensities of the pixels
- how to represent?
- as a matrix or a vector
- is affected by rotations
- by some scalar properties (mean, standard deviation)
- represents only little information


## Histograms

- the most simple patch:
- a square centered on the interest point
- properties:
- most simple: grayscale intensities of the pixels
- how to represent?
- as a matrix or a vector
- is affected by rotations
- by some scalar properties (mean, standard deviation)
- represents only little information
- by its histogram


## Histograms

- the most simple patch:
- a square centered on the interest point
- properties:
- most simple: grayscale intensities of the pixels
- is affected by global intensity fluctuations
- gradient directions
- how to represent?
- as a matrix or a vector
- is affected by rotations
- by some scalar properties (mean, standard deviation)
- represents only little information
- by its histogram


## Histograms / Intensities

- represent interest point $(x, y)$ by its $B$-dimensional intensity histogram features $\phi(x, y)$ :

$$
\begin{aligned}
\phi(x, y)_{b} & :=\left|\left\{\left(x^{\prime}, y^{\prime}\right) \in \mathcal{N}(x, y) \mid I\left(x^{\prime}, y^{\prime}\right) \in \operatorname{bin}_{b}\right\}\right|, \quad b=0, \ldots, B-1 \\
\operatorname{bin}_{b} & :=\left[\frac{b}{B} I_{\max }, \frac{b+1}{B} I_{\max }[ \right. \\
\mathcal{N}(x, y) & :=\left\{\left(x^{\prime}, y^{\prime}\right) \in[N] \times[M]| | x^{\prime}-x\left|<K,\left|y^{\prime}-y\right|<K\right\}\right.
\end{aligned}
$$

for intensities $I(x, y)$ in range $\left[0, I_{\max }\right]$.

## Histograms / Smoothed Counting

- To avoid non-continuous changes if a value crosses bin boundaries, values can be counted
- in both closest bins,
- antiproportional to their distance from the bin center

$$
\begin{gathered}
\operatorname{binc}_{b}:=\frac{b+0.5}{B} I_{\text {max }} \\
\operatorname{bin}_{b}:=\sum_{\left(x^{\prime}, y^{\prime}\right) \in \mathcal{N}(x, y)} \max \left(0,1-\frac{\left|I\left(x^{\prime}, y^{\prime}\right)-\operatorname{binc}_{b}\right|}{I_{\max } / B}\right)
\end{gathered}
$$

- sometimes called trilinear counting.


## Histograms / Gradient Directions

- represent interest point $(x, y)$ by its $B$-dimensional gradient directions histogram features $\phi(x, y)$ :

$$
\begin{aligned}
\phi(x, y)_{b} & :=\left|\left\{\left(x^{\prime}, y^{\prime}\right) \in \mathcal{N}(x, y) \mid d\left(x^{\prime}, y^{\prime}\right) \in \operatorname{bin}_{b}\right\}\right|, \quad b=0, \ldots, B-1 \\
d(x, y) & :=\tan ^{-1}\left(\left(D_{Y} * I\right)(x, y) /\left(D_{X} * I\right)(x, y)\right) \\
\operatorname{bin}_{b} & :=\left[\frac{b}{B} 2 \pi, \frac{b+1}{B} 2 \pi[ \right.
\end{aligned}
$$

## Histograms / Gradient Directions

- represent interest point $(x, y)$ by its $B$-dimensional gradient directions histogram features $\phi(x, y)$ :

$$
\begin{aligned}
\phi(x, y)_{b} & :=\left|\left\{\left(x^{\prime}, y^{\prime}\right) \in \mathcal{N}(x, y) \mid d\left(x^{\prime}, y^{\prime}\right) \in \operatorname{bin}_{b}\right\}\right|, \quad b=0, \ldots, B-1 \\
d(x, y) & :=\tan ^{-1}\left(\left(D_{Y} * I\right)(x, y) /\left(D_{X} * I\right)(x, y)\right) \\
\operatorname{bin}_{b} & :=\left[\frac{b}{B} 2 \pi, \frac{b+1}{B} 2 \pi[ \right.
\end{aligned}
$$

- variant: weight gradients by their magnitude:

$$
\phi(x, y)_{b}:=\sum_{\left(x^{\prime}, y^{\prime}\right) \in \mathcal{N}(x, y), d\left(x^{\prime}, y^{\prime}\right) \in \operatorname{bin}_{b}}\left(D_{X} * I\right)\left(x^{\prime}, y^{\prime}\right)^{2}+\left(D_{Y} * I\right)\left(x^{\prime}, y^{\prime}\right)^{2}
$$

## Histograms / Gradients / Example


[Sze11, p. 217]

## Block Descriptors

- Describe an interest point not just by features of the surrounding patch, but by the features of several neighboring patches (blocks, cells):

$$
\begin{aligned}
\phi(x, y) & :=\bigoplus_{\left(x^{\prime}, y^{\prime}\right) \in \mathcal{C}(x, y)} \phi^{\prime}\left(x^{\prime}, y^{\prime}\right) \\
\mathcal{C}(x, y) & :=\{x+c \Delta X, y+d \Delta Y \mid c, d \in\{-C, \ldots, C\}\}
\end{aligned}
$$

- Often a simple partition of a large $(2 C+1)(2 K+1) \times(2 C+1)(2 K+1)$ patch is used $(\Delta X=\Delta Y=2 K+1)$.
- Features have dimensions $(2 C+1)^{2} B$.

Note: $\left(x_{1}, \ldots, x_{N}\right) \oplus\left(y_{1}, \ldots, y_{M}\right):=\left(x_{1}, \ldots, x_{N}, y_{1}, \ldots, y_{M}\right)$ concatenation.

## Block Descriptors




Keypoint descriptor
[Low04, p. 15]

## Align Patches by the Gradient Direction of the Interest

## Point

- Extract features from the image rotated by
- the negative gradient direction at the interest point
- around the interest point
(afterwards the gradient at the interest point $(x, y)$ points towards positive $x$-direction):

$$
\begin{array}{rlrl}
\psi:= & -d(x, y) & \\
R_{\psi}\left(x^{\prime}, y^{\prime}\right): & \binom{x}{y}+\left(\begin{array}{rr}
\cos \psi & -\sin \psi \\
\sin \psi & \cos \psi
\end{array}\right)\left(\binom{x^{\prime}}{y^{\prime}}-\binom{x}{y}\right) \\
I_{\mathrm{bi}}(x, y): & (1-(x-\lfloor x\rfloor))(1-(y-\lfloor y\rfloor)) & & I(\lfloor x\rfloor,\lfloor y\rfloor) \\
& +(x-\lfloor x\rfloor)(1-(y-\lfloor y\rfloor)) & I(\lceil x\rceil,\lfloor y\rfloor) \\
& +(1-(x-\lfloor x\rfloor))(y-\lfloor y\rfloor) & I(\lfloor x\rfloor,\lceil y\rceil) \\
& +(x-\lfloor x\rfloor)(y-\lfloor y\rfloor) & I(\lceil x\rceil,\lceil y\rceil)
\end{array}
$$

(bilinear interpolation)

## SIFT descriptors

- patches:
- extract from the scaled image the interest point has been detected on
- align patch by the gradient direction of the interest point
- $16 \times 16$, partitioned into 16 blocks a $4 \times 4$
- block features:
- gradient directions
- weighted by a Gaussian of the distance to the interest point
- block feature aggregation:
- smoothly counted histograms
- 8 bins
- $\rightsquigarrow$ feature vector $\phi \in \mathbb{R}^{128}$
- normalization in 3 steps:

$$
\phi_{i}^{\prime}:=\phi_{i} /\|\phi\|_{2}, \quad \phi_{i}^{\prime \prime}:=\min \left(0.2, \phi_{i}^{\prime}\right), \quad \phi_{i}^{\prime \prime \prime}:=\phi_{i}^{\prime \prime} /\left\|\phi^{\prime \prime}\right\|_{2}
$$

## Image Descriptors

To describe a whole image (not just a patch), two main approaches are used:

1. Concatenate patch descriptors of equally spaced "interest points" 1.1 e.g., used in Histograms of Oriented Gradients (HoG)

## Image Descriptors

To describe a whole image (not just a patch), two main approaches are used:

1. Concatenate patch descriptors of equally spaced "interest points" 1.1 e.g., used in Histograms of Oriented Gradients (HoG)
2. Bag of words descriptors:
2.1 compute interest points and their descriptors for a set of images
2.2 discretize the descriptors

- e.g., clustering in $K$ clusters using k-means
2.3 represent each image by the $K$ cluster frequencies of their interest point descriptors


## Histograms of Oriented Gradients (HoG)



Figure 13.17 HOG descriptor. a) Original image. b) Gradient orientation, quantized into nine bins from 0 to $180^{\circ}$. c) Gradient magnitude. d) Cell descriptors are 9D orientation histograms that are computed within $6 \times 6$ pixel regions. e) Block descriptors are computed by concatenating $3 \times 3$ blocks of cell descriptors. The block descriptors are normalized. The final HOG descriptor consists of the concatenated block descriptors.
[Pri12, p. 343]

## Outline

## 1. Smoothing, Image Derivatives, Convolutions

2. Edges, Corners, and Interest Points
3. Image Patch Descriptors
4. Interest Point Matching

## 5. A Simple Application: Image Stitching

## Settings, Assumptions, Distances

Two settings:

- match interest points in different scenes
- goal: detect similar objects (object identification)
- coordinates of the points do not matter

$$
d\left(\binom{x_{1}}{y_{1}},\binom{x_{2}}{y_{2}}\right):=d^{\prime}\left(\phi\left(x_{1}, y_{1}\right), \phi\left(x_{2}, y_{2}\right)\right)=\left\|\phi\left(x_{1}, y_{1}\right)-\phi\left(x_{2}, y_{2}\right)\right\|_{2}
$$

$$
\begin{aligned}
d\left(\binom{x_{1}}{y_{1}},\binom{x_{2}}{y_{2}}\right) & :=\alpha d^{\prime}\left(\binom{x_{1}}{y_{1}},\binom{x_{2}}{y_{2}}\right)+\beta d^{\prime}\left(\phi\left(x_{1}, y_{1}\right), \phi\left(x_{2}, y_{2}\right)\right) \\
& =\alpha\left\|\binom{x_{1}}{y_{1}}-\binom{x_{2}}{y_{2}}\right\|_{2}+\beta\left\|\phi\left(x_{1}, y_{1}\right)-\phi\left(x_{2}, y_{2}\right)\right\|_{2}
\end{aligned}
$$

## Settings, Assumptions, Distances

## Two settings:

- match interest points in different scenes
- goal: detect similar objects (object identification)
- coordinates of the points do not matter

$$
d\left(\binom{x_{1}}{y_{1}},\binom{x_{2}}{y_{2}}\right):=d^{\prime}\left(\phi\left(x_{1}, y_{1}\right), \phi\left(x_{2}, y_{2}\right)\right)=\left\|\phi\left(x_{1}, y_{1}\right)-\phi\left(x_{2}, y_{2}\right)\right\|_{2}
$$

- match interest points in two views of the same scene
- goal: detect corresponding points in different views of the same scene (required for SLAM)
- coordinates of corresponding points also should be close, e.g.,

$$
\begin{aligned}
& d\left(\binom{x_{1}}{y_{1}},\binom{x_{2}}{y_{2}}\right):=\alpha d^{\prime}\left(\binom{x_{1}}{y_{1}},\binom{x_{2}}{y_{2}}\right)+\beta d^{\prime}\left(\phi\left(x_{1}, y_{1}\right), \phi\left(x_{2}, y_{2}\right)\right) \\
&=\alpha\left\|\binom{x_{1}}{y_{1}}-\binom{x_{2}}{y_{2}}\right\|_{2}+\beta\left\|\phi\left(x_{1}, y_{1}\right)-\phi\left(x_{2}, y_{2}\right)\right\|_{2} \\
&
\end{aligned}
$$

## Simple methods

To match two sets $P$ and $Q$ of interest points:

- match interest points by distance threshold

$$
p \sim q: \Leftrightarrow d(p, q)<d_{\max }, \quad p \in P, q \in Q
$$

- distance threshold $d_{\text {max }}$ can be estimated from known matches and non-matches


## Simple methods

To match two sets $P$ and $Q$ of interest points:

- match interest points by distance threshold

$$
p \sim q: \Leftrightarrow d(p, q)<d_{\max }, \quad p \in P, q \in Q
$$

- distance threshold $d_{\text {max }}$ can be estimated from known matches and non-matches
- match interest points by nearest neighbor

$$
p \sim q: \Leftrightarrow q=\underset{q \in Q}{\arg \min } d(p, q)
$$

## Nearest Neighbor Distance Ratio

- match interest points by nearest neighbor distance ratio (NNDR)
$p \sim q: \Leftrightarrow i) q=\underset{q \in Q}{\arg \min } d(p, q)$ and
ii) $\operatorname{NNDR}(p, q):=\frac{d(p, q)}{d\left(p, q^{\prime}\right)}<\operatorname{NNDR}_{\text {min }}, \quad q^{\prime}:=\underset{q^{\prime} \in Q \backslash\{q\}}{\arg \min } d\left(p, q^{\prime}\right)$



## Comparison of Different Descriptors \& Matchings

a) fixed threshold:

[Sze11, p. 229]

## Comparison of Different Descriptors \& Matchings

b) nearest neighbor:

[Sze11, p. 229]

## Comparison of Different Descriptors \& Matchings

c) nearest neighbor distance ratio:

[Sze11, p. 229]

## Mutual Nearest Neighbors

- match interest points if they mutually are nearest neighbors

$$
\begin{gathered}
p \sim q: \Leftrightarrow i) q=\underset{q \in Q}{\arg \min } d(p, q) \text { and } \\
\text { ii) } p=\underset{p \in P}{\arg \min } d(p, q)
\end{gathered}
$$

- also for more than two views $P_{1}, P_{2}, \ldots, P_{V}$ (called closed chains)
$\left(p_{1}, p_{2}, \ldots, p_{V}\right)$ corresponding tuple

$$
\begin{aligned}
& : \Leftrightarrow \quad \text { i) } p_{v+1}=\underset{q \in P_{v+1}}{\arg \min } d\left(p_{v}, q\right), \quad v=1, \ldots, V-1 \text { an } \\
& \quad \text { ii) } p_{1}=\underset{q \in P_{v}}{\arg \min } d\left(p_{1}, q\right)
\end{aligned}
$$

## Outline

## 1. Smoothing, Image Derivatives, Convolutions

2. Edges, Corners, and Interest Points
3. Image Patch Descriptors
4. Interest Point Matching

## 5. A Simple Application: Image Stitching

## Image Stitching

- join several images depicting overlapping parts of the same real scene to one large image


## Image Stitching

- join several images depicting overlapping parts of the same real scene to one large image
- algorithm:

1. detect interest points in all images and extract their descriptors
2. match interest points between every two images
3. form a tree linking the best matching image pairs
4. estimate a similarity transform between each two such images
5. transform all images to joint coordinates
6. average overlapping image regions

## Image Stitching

- join several images depicting overlapping parts of the same real scene to one large image
- algorithm:

1. detect interest points in all images and extract their descriptors
2. match interest points between every two images
3. form a tree linking the best matching image pairs
4. estimate a similarity transform between each two such images
5. transform all images to joint coordinates

6 . average overlapping image regions

- also called panography


## Image Stitching / Example


[Sze11, p. 312]

## Image Stitching / Different Transforms


(a) translation [2 dof]

(b) affine [6 dof]

(c) perspective [8 dof]

(d) 3 D rotation $[3+\mathrm{d}$

## Image Stitching / Example



## Summary

- Small intensity fluctuations can be damped by smoothing, intensity changes can be captured by image derivatives, both being convolutions.
- Interest points are found as maxima of an interestingness measure,
- gradient magnitude, Laplacian of Gaussian (LoG), Different of two Gaussians (DoG)
- Harris corners:
- large eigenvalues of the Hessian
- can be approximated efficiently: $\operatorname{det} H-\alpha(\text { trace } H)^{2}$
- SIFT:
- detected interest points at different scale
- several further tweaks
- non-maximum suppressions: ignore large values in the vicinity of a maximum


## Summary (2/3)

- Interest points are characterized by local image information (descriptors)
- Descriptors often describe several patches (blocks/cells)
- Patches are described by histograms
- Histograms usually do not count pixel intensities, but gradient directions
- Descriptors sometimes
- align patches with the orientation of the gradient at the interest point
- weight gradient directions by their
- gradient magnitude and/or
- distance of the location to the interest point
- Common descriptors:
- SIFT descriptors, Histogram of Gradients (HoG)


## Summary (3/3)

- Whole images can be described two ways:
- by the descriptors on a fixed grid of "interest points"
- by the cluster frequencies of descriptors of variably located interest points
Both is useful, e.g. for image classification.
- Interest points are matched by their descriptors
- for geometric tasks: also by their positons
- To match interest points, nearest neighbors are used
- with a maximal distance threshold to avoid wrong matches e.g. of points occluded in one view
- Nearest Neighbor Distance Ratio
- mutual nearest neighbors, closed chains in multiple views.
- Corresponding points can be used for
- image stitching
- SLAM, camera auto-calibration, ...


## Further Readings

- Interest points and patch descriptors: [Pri12, ch. 13], [Sze11, ch. 4].
- Image stitching: [Sze11, ch. 9].


## References

David G. Lowe.
Distinctive image features from scale-invariant keypoints.
International journal of computer vision, 60(2):91-110, 2004.
Simon JD Prince.
Computer vision: models, learning, and inference.
Cambridge University Press, 2012.
Richard Szeliski.
Computer vision: algorithms and applications.
Springer Science \& Business Media, 2011.

