– SSDA 2019 / Islamabad –
Aug 20, 2019

AMIGO – Automatic Indexing of Lecture Footage

Prof. Dr. Adrian Ulges
DCSM Department
RheinMain University of Applied Sciences

Outline

E-Learning

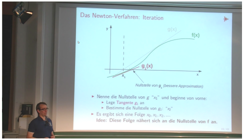
E-learning

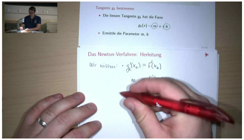
+ choose your learning time
+ choose your learning location
+ choose your learning speed
+ choose your learning depth

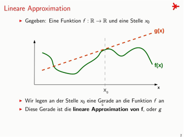
E-Learning

E-learning

- + choose your learning time
- + choose your learning location
- + choose your learning speed
- + choose your learning depth







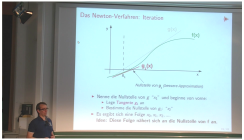
Educational Videos are a Key Driver

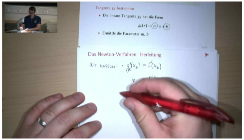
- Lecture recordings, screencasts, webcasts, ...
- coursera, Khan Academy, udacity, ...

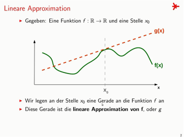
E-Learning

E-learning

+ choose your learning time
+ choose your learning location
+ choose your learning speed
+ choose your learning depth







Educational Videos are a Key Driver

- Lecture recordings, screencasts, webcasts, ...
- coursera, Khan Academy, udacity, ...
- **Challenge: interaction is limited!**

Motivation

Learning requires **interaction**

- navigation (*where in the video does section 3 start?*)
- fine-grain access (*where can I find Example X?*)
- storage and reorganisation (*can I copy text from the video?*)
- exploration (*where can I find additional material?*)

Motivation

Learning requires **interaction**

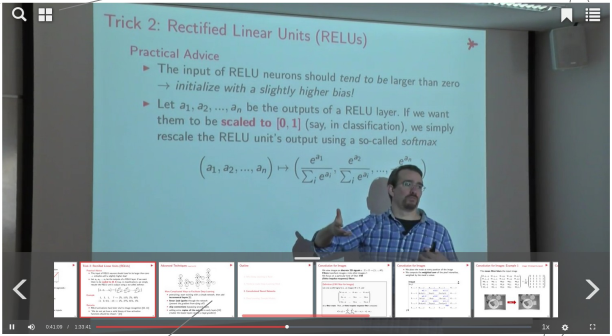
- navigation (*where in the video does section 3 start?*)
- fine-grain access (*where can I find Example X?*)
- storage and reorganisation (*can I copy text from the video?*)
- exploration (*where can I find additional material?*)

AMIGO Video Platform → https://video.cs.hs-rm.de

Text Search : find topics at the exact second

Display current slide (high-res)

bookmarking



navigate between slides

change **speed + resolution**

Questions? → amigo@hs-rm.de

developed at HSRM

AMIGO: Mission Statement

Rich Interaction with Videos

… just like with (digital) documents

- navigate between pages
- text search
- hyperlinks

AMIGO: Mission Statement

Rich Interaction with Videos

... just like with (digital) documents

- navigate between pages
- text search
- hyperlinks

Key Features

- **automatic slide matching**
  - video = pixels
  - slides = PDF

AMIGO: Mission Statement

Rich Interaction with Videos

... just like with (digital) documents

- navigate between pages
- text search
- hyperlinks

Key Features

- **automatic slide matching**
  - video = pixels
  - slides = PDF
- **automatic wikification**
  - find interesting phrases (“convolutional neural network”)
  - link them with Wikipedia

AMIGO: Mission Statement

Rich Interaction with Videos

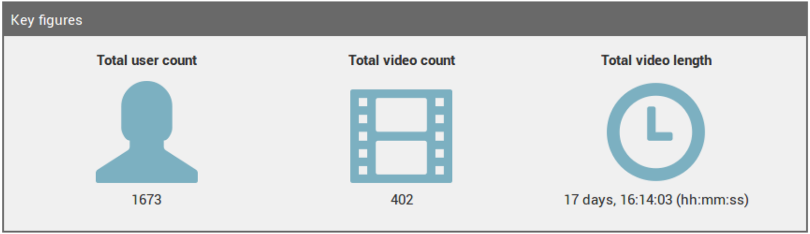
... just like with (digital) documents

- navigate between pages
- text search
- hyperlinks

Key Features

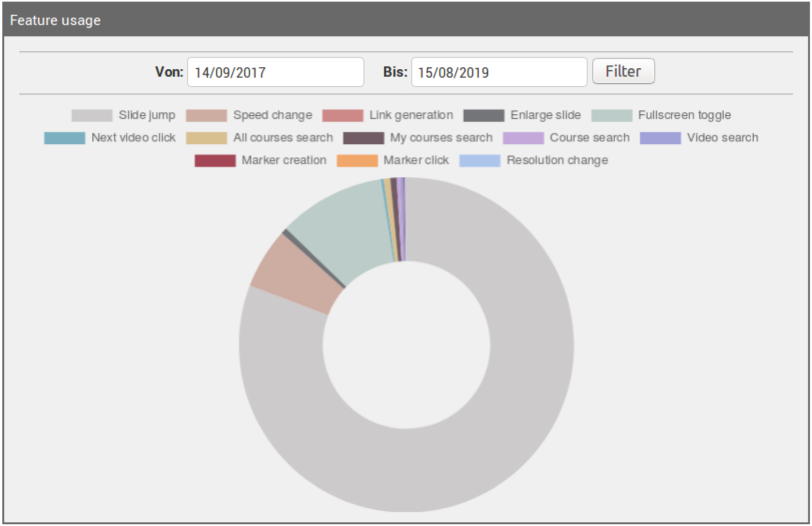
- **automatic slide matching**
  - video = pixels
  - slides = PDF
- **automatic wikification**
  - find interesting phrases ("convolutional neural network")
  - link them with Wikipedia
- **learning analytics**
  - anynomous tracking of user actions
  - which video passages do students watch?
  - which terms do students search for?

AMIGO: Statistics[1]

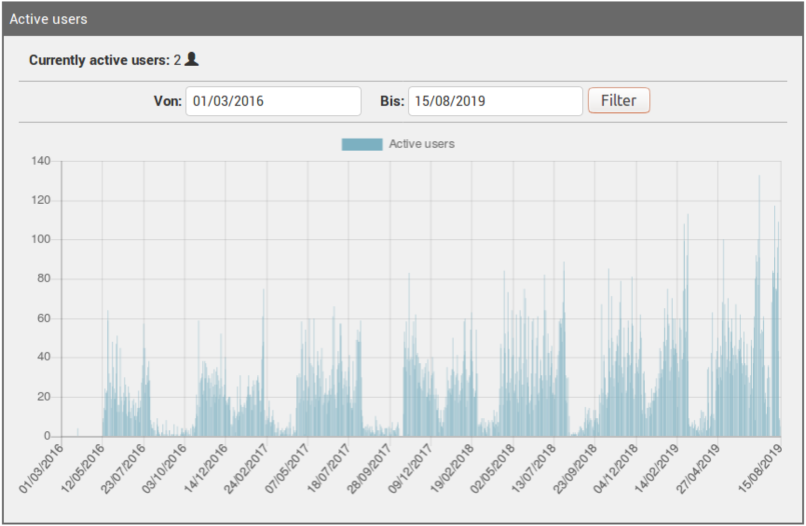


[1] status: 2019/08

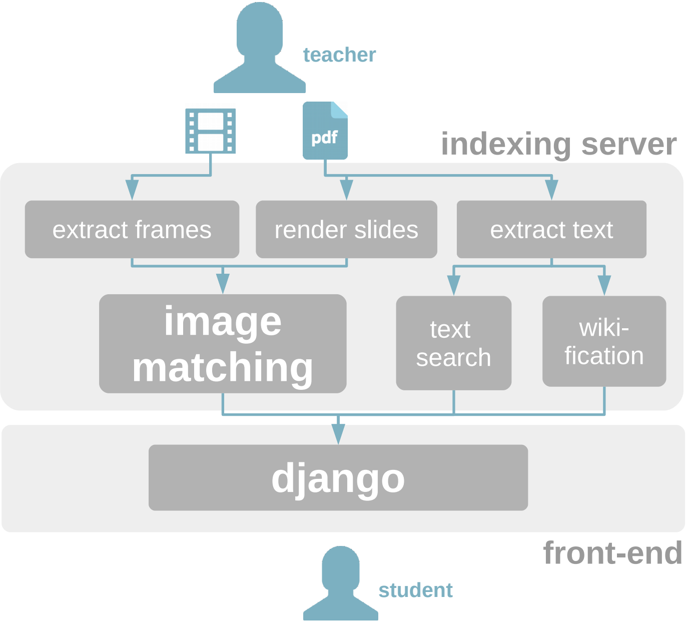
AMIGO: Statistics (cont'd)



AMIGO: Statistics (cont'd)



AMIGO Workflow



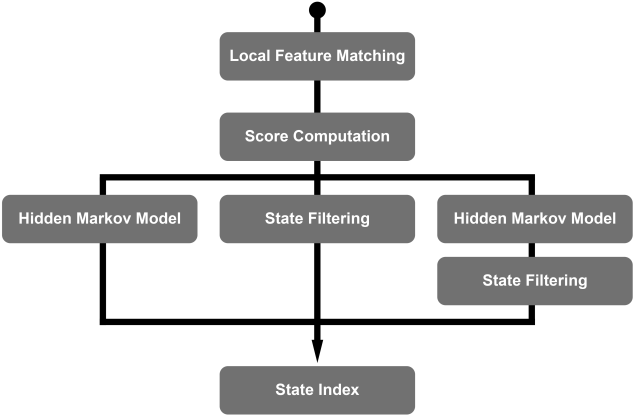
Outline

Image Matching in AMIGO

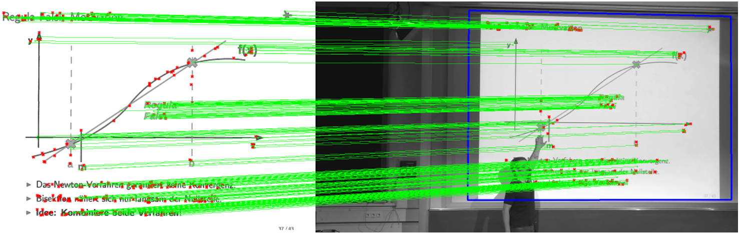
AMIGO matches slides in the lecture PDF with frames in the video

Two Main Steps

1. Keypoint Matching
2. Temporal Filtering



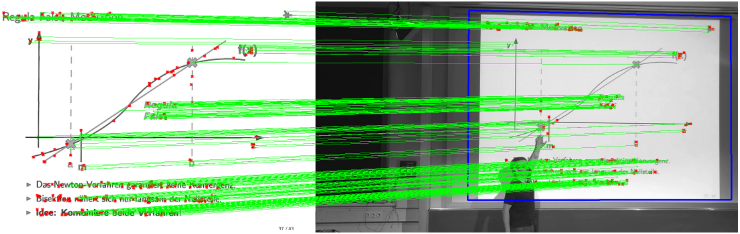
Keypoint Matching



- Video frames $\mathcal{F} = \{f_1, \ldots, f_m\}$ are sampled (1 per second)
- Slide images $\mathcal{S} = \{s_1, ..., s_n\}$ are rendered (1 per slide)

Goal: Compute an **indexing**: a **mapping** from $\mathcal{F}$ to $\mathcal{S} \cup \{s_0\}$
(s_0 = no slide visible)

Keypoint Matching



- Video frames $\mathcal{F} = \{f_1, \ldots, f_m\}$ are sampled (1 per second)
- Slide images $\mathcal{S} = \{s_1, ..., s_n\}$ are rendered (1 per slide)

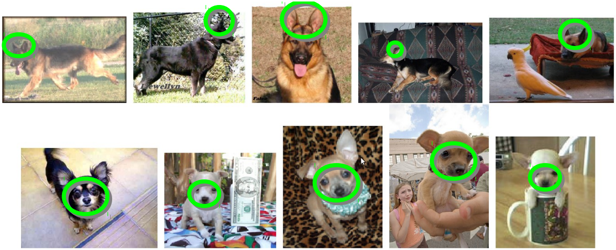
Goal: Compute an **indexing**: a **mapping** from $\mathcal{F}$ to $\mathcal{S} \cup \{s_0\}$ *(s_0 = no slide visible)*

1. Match **SIFT features** between $\mathcal{S}$ and $\mathcal{F}$.
2. Improve the match quality using several **filters** *(NN-ratio of descriptor distance, homography estimation, ...)*

Outline

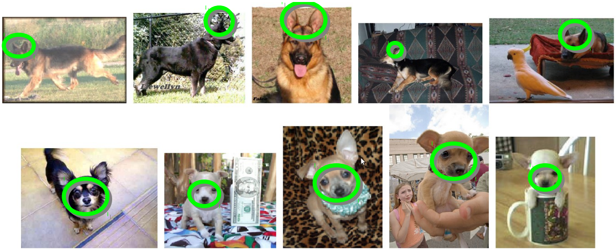
Local Features: Motivation[2]

Key Idea: Even when images from the same class are not **globally** similar, they share certain **local characteristics**



Local Features: Motivation[2]

Key Idea: Even when images from the same class are not **globally** similar, they share certain **local characteristics**

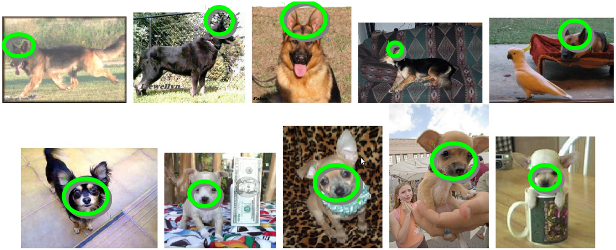


Approach 1: Hand-engineered Local Features (here)

- state-of-the-art until 2011 *(and still used frequently today)*
- **SIFT**, SURF, HoG, Canny, ORB, ...

Local Features: Motivation[2]

Key Idea: Even when images from the same class are not **globally** similar, they share certain **local characteristics**



Approach 1: Hand-engineered Local Features (here)

- state-of-the-art until 2011 *(and still used frequently today)*
- **SIFT**, SURF, HoG, Canny, ORB, ...

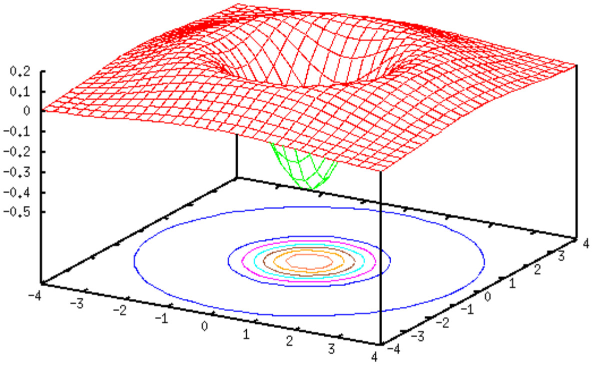
Approach 2: Learn Local Features

- state-of-the-art since 2011
- Convolutional Neural Networks (CNNs)

BLOB Detection: Example

How do we detect blob ... **at different scales**?

The DoG-Filter: Illustration image: [3]



- The DoG filter approximates the so-called **Mexican Hat** (aka "Laplacian-of-Gaussians") operator
- The DoG filter detects **blobs** *(dark regions surrounded by a bright background)*

Feature Detection: Scale Invariance

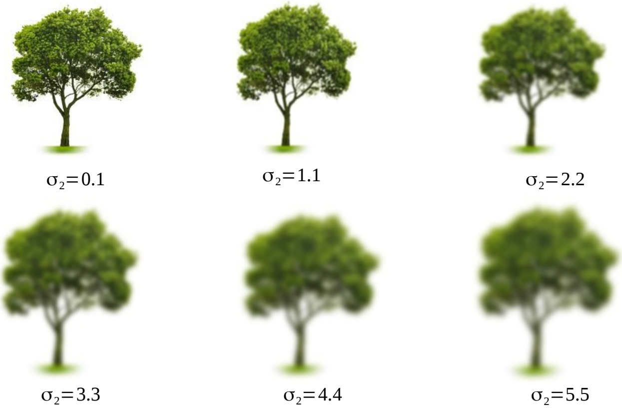
- Modern feature detectors come with a free **scale parameter**
- For DoG: the scale σ_2 *(from which we compute* σ_1*)*

Feature Detection: Scale Invariance

- Modern feature detectors come with a free **scale parameter**
- For DoG: the scale σ_2 *(from which we compute* σ_1*)*
- This parameter determines if our detector localizes **fine, small** structures or **coarse, wide-spread** structures

Feature Detection: Scale Invariance

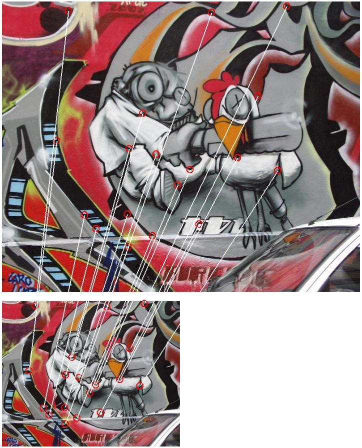
- Modern feature detectors come with a free **scale parameter**
- For DoG: the scale σ_2 *(from which we compute σ_1)*
- This parameter determines if our detector localizes **fine, small** structures or **coarse, wide-spread** structures



Outline

Local Features: Matching image: [1]

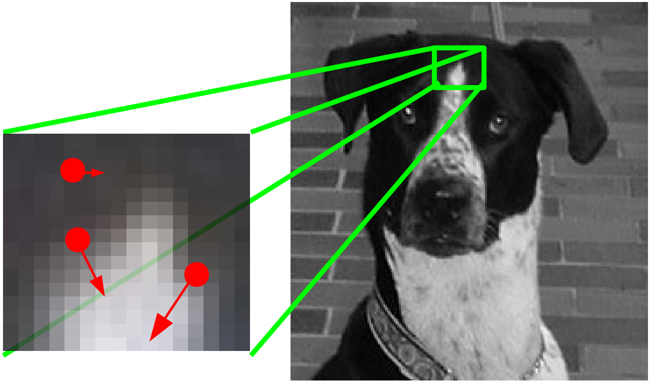
After extracting local features, we *match* them to recognize objects



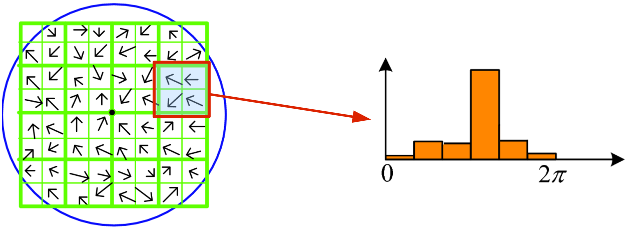
The Gradient: Properties

Remarks

- The gradient always points into the direction of the **strongest increase in intensity**.
- The gradient's norm $||s(x, y)||$ corresponds to the **strength of the edge**.

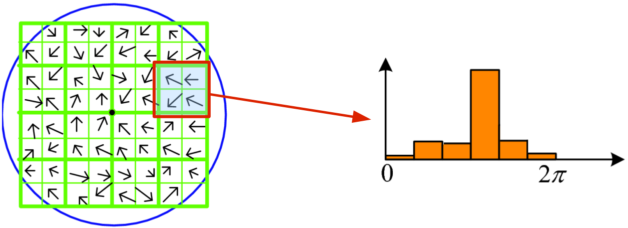


SIFT Features image: [5]



2. Description by Gradient Histograms
   - Subdivide the (normalized) ROI into 4×4 windows.
   - For each window, store a **normalized histogram** of the 8 (discretized) gradient orientations.

SIFT Features image: [5]



2. Description by Gradient Histograms
   - Subdivide the (normalized) ROI into 4×4 windows.
   - For each window, store a **normalized histogram** of the 8 (discretized) gradient orientations.
   - **Concatenate** the 4×4 histograms (each 8-dimensional) into a 128-dimensional local feature vector

Outline

Hidden Markov Model (HMM)

Simple Idea

Hidden Markov Model (HMM)

Simple Idea

For each frame, pick the highest-scored slide → error-prone ☹

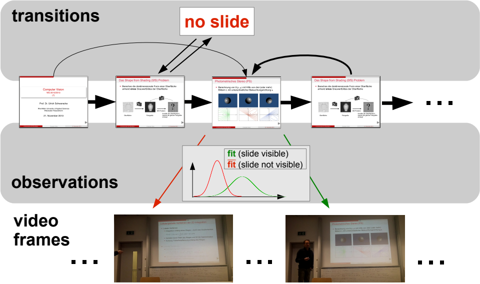
Hidden Markov Model (HMM)

Simple Idea

For each frame, pick the highest-scored slide → error-prone ☹

Idea: Employ **reading order** of material!

- HMM: For each frame, infer a state (slide) based on two constraints
    1. Transistions between certain slides are more likely
    2. Slides should match the video content well



Outline

State Filtering

Observation

- There are still short subsegments with instable recognitions

  (slide 7 → slide 18 → slide 7 → ...)

State Filtering

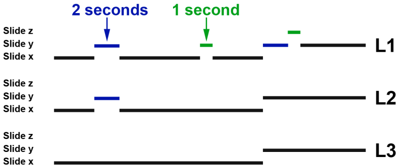
Observation

- There are still short subsegments with instable recognitions

  (slide 7 → slide 18 → slide 7 → ...)

Approach: Heuristic Filtering

/* for segment length up to τ */
for L in 1,...,τ **do**
  /* iterate over all segments s */
  for each segment **do**
    if segment duration $\leq L$ seconds **then**
      merge the segment with its predecessor
    end if
  end for
end for



Outline

Experiments: Recognition Results

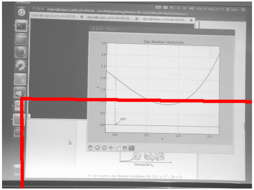
Indexing at 1 fps → 12,164 frame-slide pairs

- Manual annotation for each frame-slide pair
- Two different quality indicators
  - Percentage of frames with correctly recognized slides (**state accuracy (SA)**)
  - Correctness of slide transitions (**Jaccard index (JI)**, $J(\mathcal{T}, \mathcal{T}') = \frac{|\mathcal{T} \cap \mathcal{T}'|}{|\mathcal{T} \cup \mathcal{T}'|}$ with true transitions $\mathcal{T}$ and transitions recognized by AMIGO $\mathcal{T}'$)

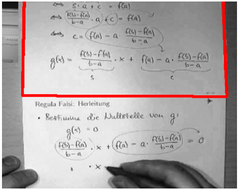
Course	Topic	baseline		homography valid.		hom.v. & HMM		final	
		JI	SA	JI	SA	JI	SA	JI	SA
CV	SfS	1.94	59.58	28.18	96.75	73.75	98.24	93.02	98.96
Analysis	Bisection	2.65	66.45	26.32	91.67	45.45	92.39	64.71	96.31
Analysis	Newton	4.81	71.60	16.07	93.45	45.00	95.39	60.00	96.96
Analysis	Motivation	4.35	76.97	33.33	95.76	77.78	97.37	100.00	99.18
Analysis	Regula Falsi	3.30	75.16	26.98	85.86	47.37	86.32	69.23	87.74
Analysis	Taylor series	5.89	88.33	8.33	88.85	15.19	90.99	73.33	91.18
average		**3.82**	**73.02**	**23.20**	**92.05**	**50.76**	**93.45**	**76.72**	**95.05**

Experiments: Error Inspection

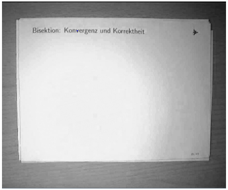
We found 8 incorrect subsequences, caused by 4 different sources of error:



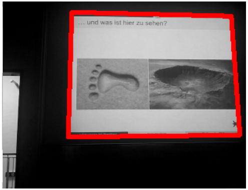
partial occlusion



redundant content



lack of texture



missing content

The End

References I

[1] Affine Covariant Features Dataset.
http://www.robots.ox.ac.uk/~vgg/research/affine/ (retrieved: Oct 2016).

[2] picture shared by Christoph Lampert.
contact: http://pub.ist.ac.at/~chl/.

[3] Wang, R.: Computer Image Processing and Analysis (E161) Course (Harvey Mudd College).
http://fourier.eng.hmc.edu/e161/lectures/gradient/node8.html (retrieved: Oct 2016).

[4] Yes, this is Megan Fox.
like, everywhere on the internet... (retrieved: Oct 2016).

[5] D. G. Lowe.
Distinctive image features from scale-invariant keypoints.
Int. J. Comput. Vision, 60(2):91–110, 2004.