Classic Video Datasets and Algorithms

Qasim Nadeem, Divya Thuremella

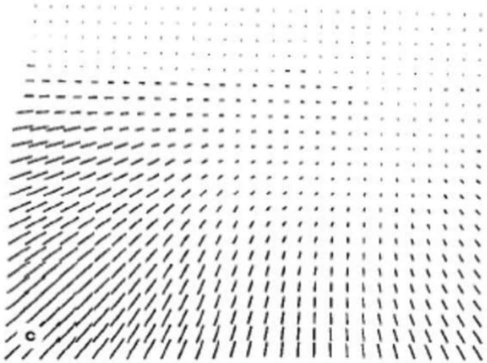
Overview

- Background
- Action Recognition Dense Trajectories Paper
    - Main Idea
    - Ways to Track
    - Dense Trajectories
    - Descriptors
- Datasets
    - KTH
    - UCF 101
    - Hollywood-2
    - HMDB
- Evaluation

Background: Optical Flow (and Lukas Kanade Tracking)

Optical Flow

- 2D vector field describing apparent motion in images



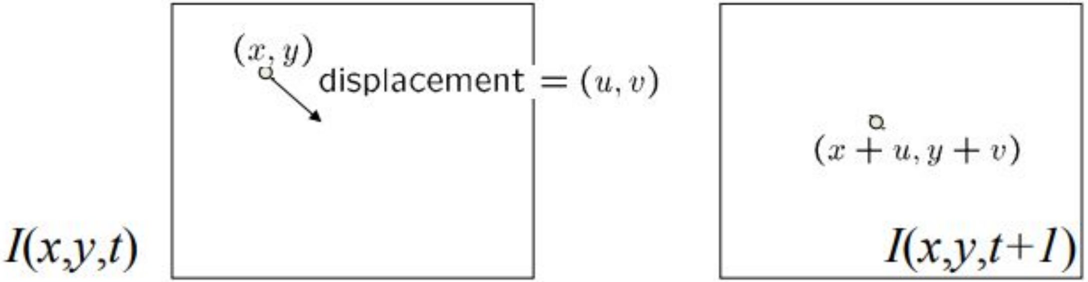
$u(x, y)$ Horizontal component

$v(x, y)$ Vertical component

Lucas Kanade Object Tracker

- Key assumptions:
  - **Brightness constancy:** projection of the same point looks the same in every frame (uses SSD as metric)
  - **Small motion:** points do not move very far (from guessed location)
  - **Spatial coherence:** points move in some coherent way (according to some parametric motion model)
    - For this example, assume whole object just translates in (u,v)

Lukas Kanade Object Tracker



- Brightness Constancy Equation:

$$I(x,y,t) = I(x+u,y+v,t+1)$$

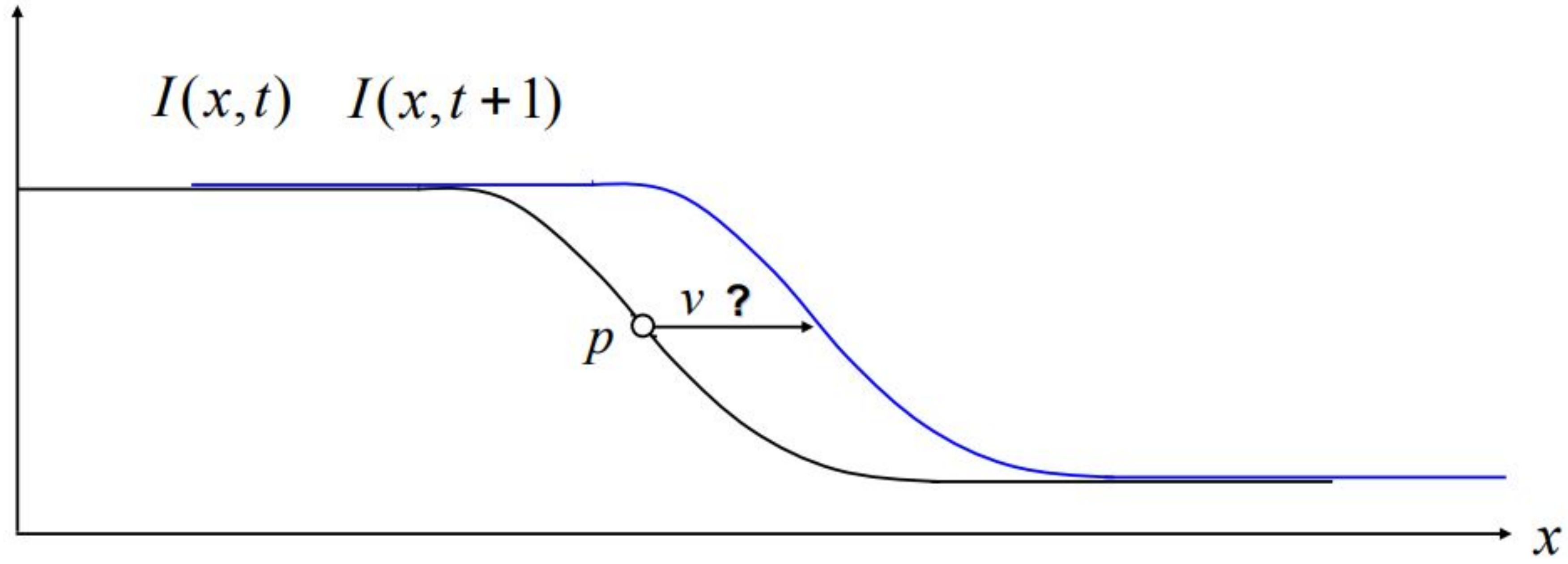
Take Taylor expansion of $I(x+u, y+v, t+1)$ at (x,y,t) to linearize the right side:

$$I(x+u, y+v, t+1) \approx I(x, y, t) + \overset{\text{Image derivative along x}}{\boxed{I_x}} \cdot u + I_y \cdot v + \overset{\text{Difference over frames}}{\boxed{I_t}}$$

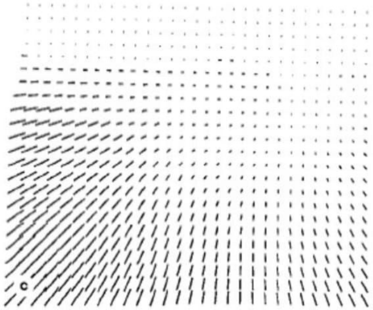
$$I(x+u, y+v, t+1) - I(x, y, t) = + I_x \cdot u + I_y \cdot v + I_t$$

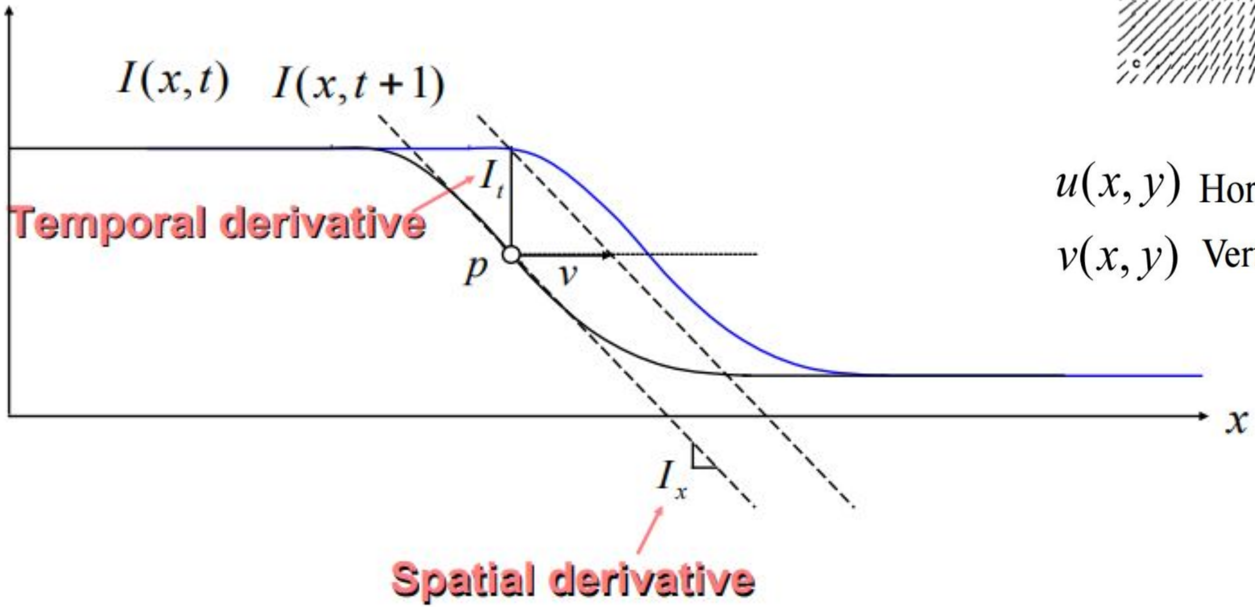
Hence,

$$I_x \cdot u + I_y \cdot v + I_t \approx 0 \quad \rightarrow \nabla I \cdot [u \;\; v]^T + I_t = 0$$


$I(x,t)$
$I(x,t+1)$
v ?
p
x

Calculating Optical Flow





$u(x,y)$ Horizontal component

$v(x,y)$ Vertical component

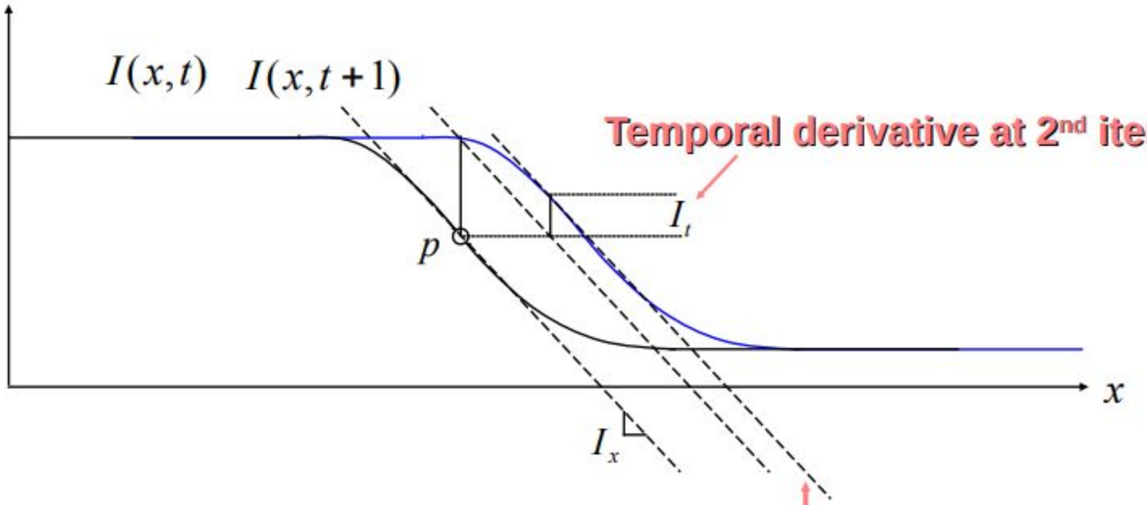
$$I_x = \frac{\partial I}{\partial x}\Big|_t \qquad I_t = \frac{\partial I}{\partial t}\Big|_{x=p} \qquad \Rightarrow \qquad \vec{v} \approx - \frac{I_t}{I_x}$$

Assumptions:

- **Brightness constancy**
- **Small motion**

Lukas Kanade: Find New Position from Optical Flow

Iterating helps refining the velocity vector



Temporal derivative at 2nd iteration

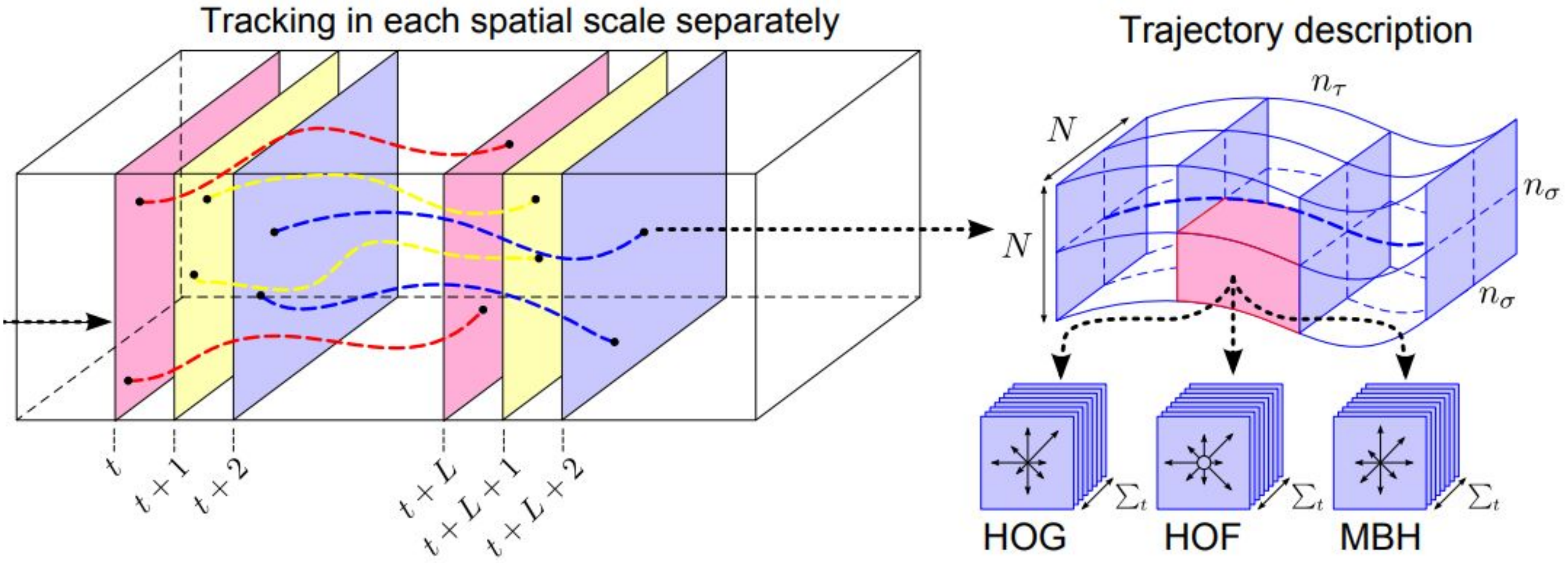
Can keep the same estimate for spatial derivative

$$\vec{v} \leftarrow \vec{v}_{previous} - \frac{I_t}{I_x}$$

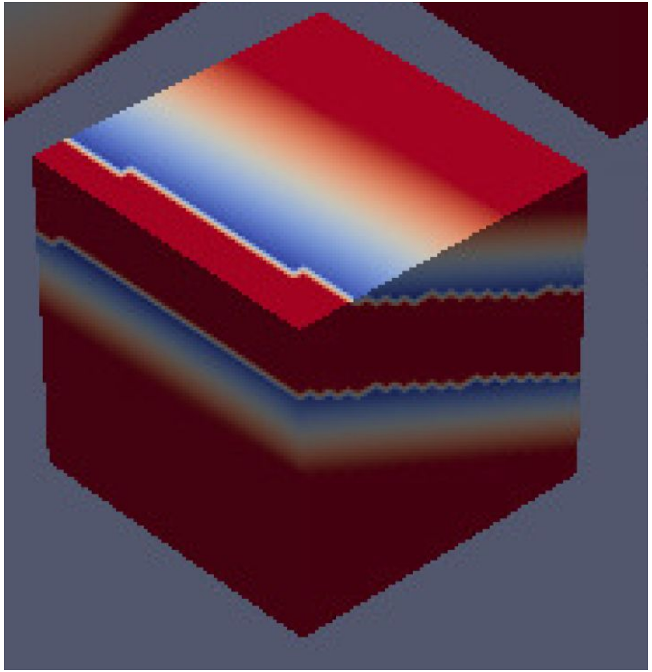
Converges in about 5 iterations

Paper: Action Recognition by Dense Trajectories

Main Idea: Videos as Trajectories



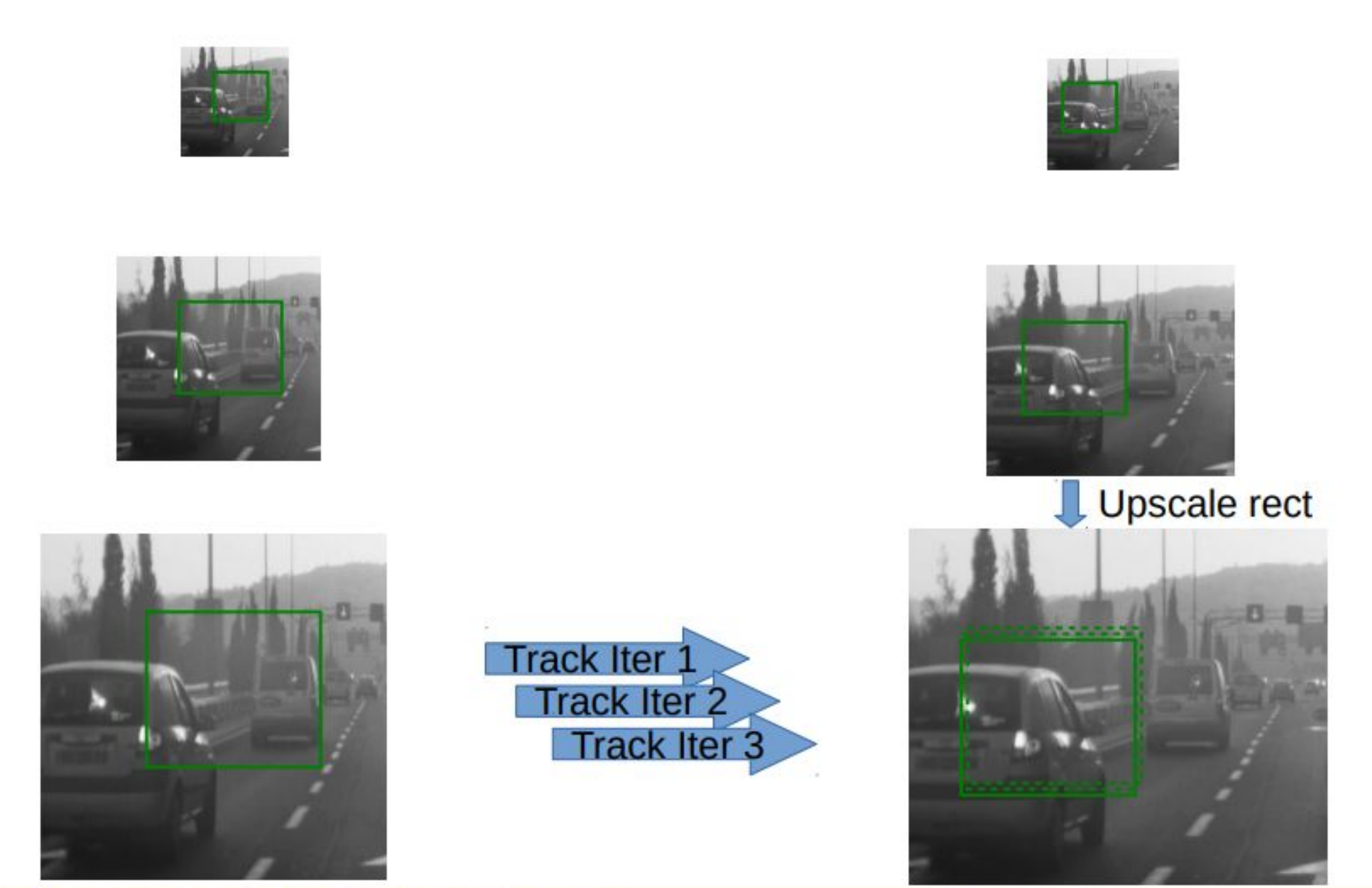
Main Idea: Videos as Trajectories



Ways to Track

1) **Lukas Kanade Tracker**
    a) Baseline
2) Find SIFT features and match between frames
    a) Too expensive
3) Dense Trajectories (proposed method)

Wa



1)

ared

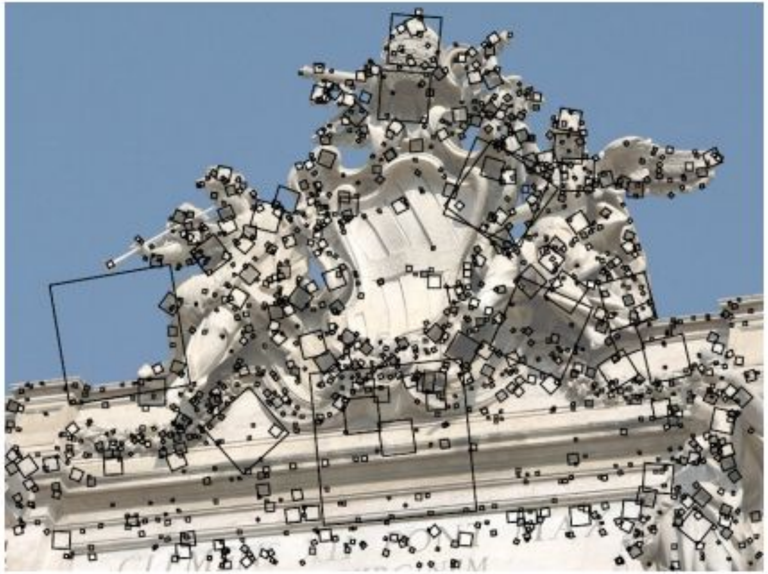
2)

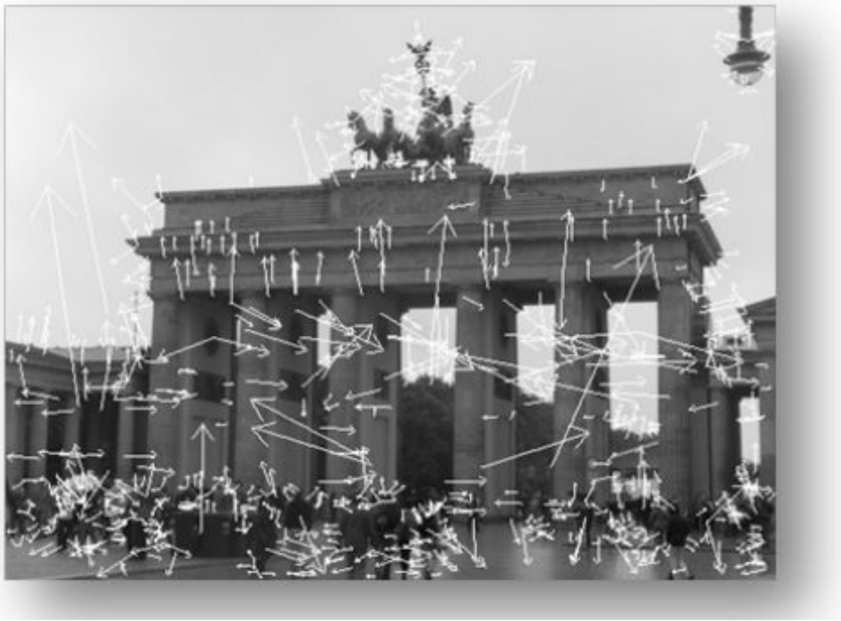
3)

Ways to Track

1) Lukas Kanade Tracker
    a) Baseline
2) **Find SIFT features and match between frames**
    a) Too expensive
3) Dense Trajectories (proposed method)

Ways to Track





Ways to Track

1) Lukas Kanade Tracker
   a) Baseline
2) Find SIFT features and match between frames
   a) Too expensive

3) Dense Trajectories (proposed method)

Ways

1) Lukas K

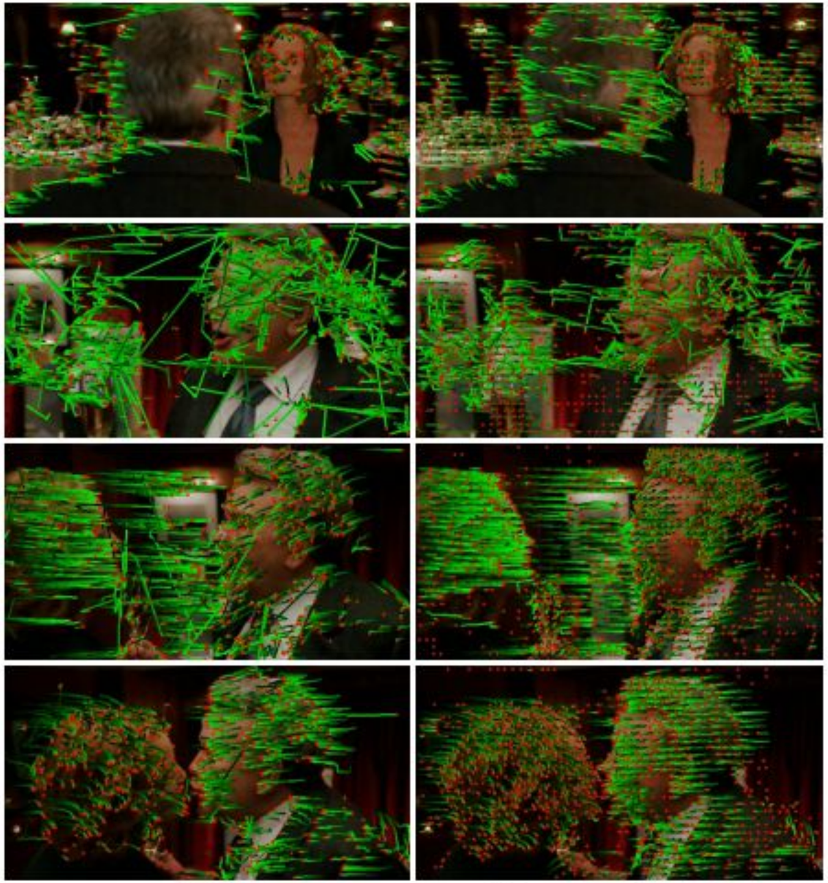
   a) Ba s being compared

      to i

2) Find SI ame

   a) To tures

3) Dens nethod) <--Best Method



Dense Trajectories

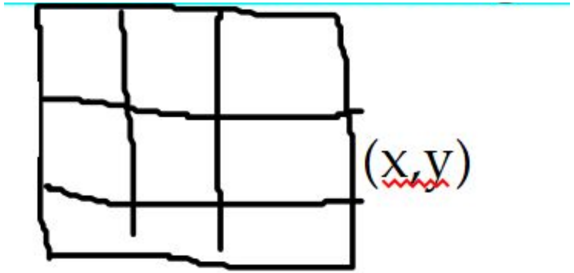
- Image → WxW grid, tracked in each scale
- If not connected to prev track, start a new one
- update each point in box with median of all points in that box

Dense Trajectories

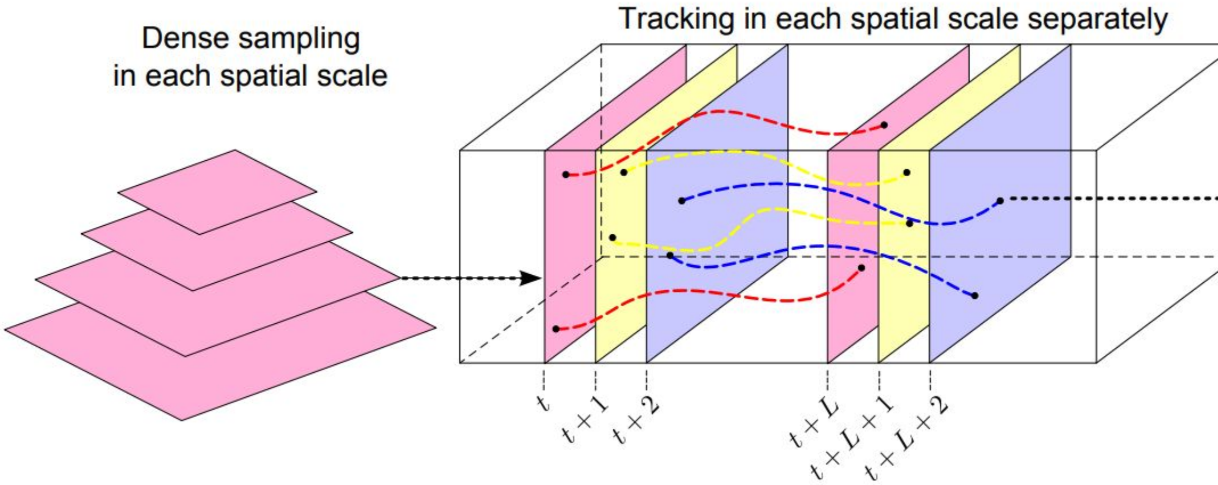
- **Image → WxW grid, tracked in each scale**
- If not connected to prev track, start a new one
- update each point in box with median of all points in that box

Dense Trajectories

- Image → WxW grid, tracked in each scale



Each box at each scale is tracked separately

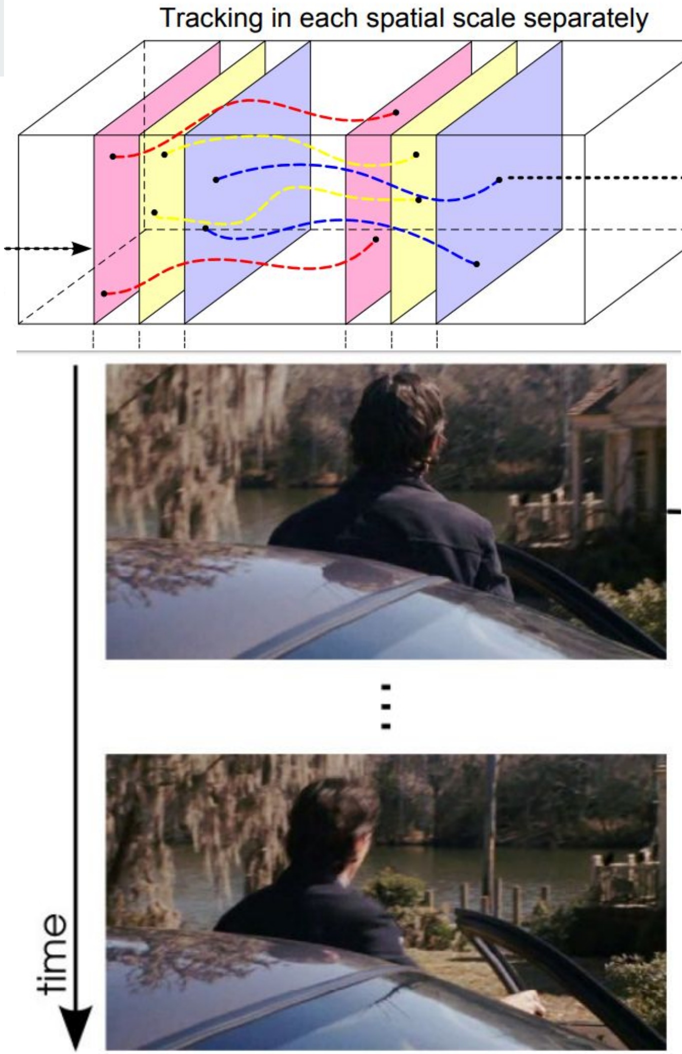


Dense Trajectories

- Image → WxW grid, tracked in each scale
- **If not connected to prev track, start a new one**
- update each point in box with median of all points in that box

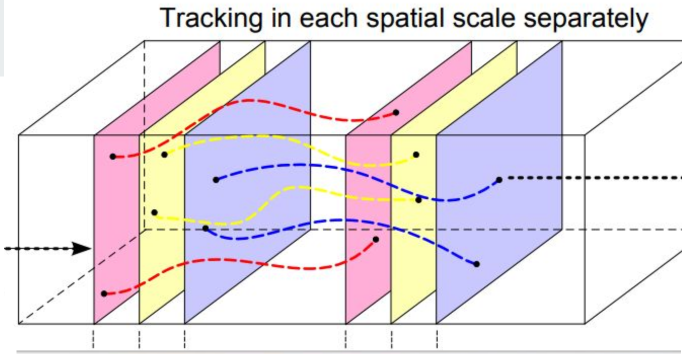
Dense Trajectories

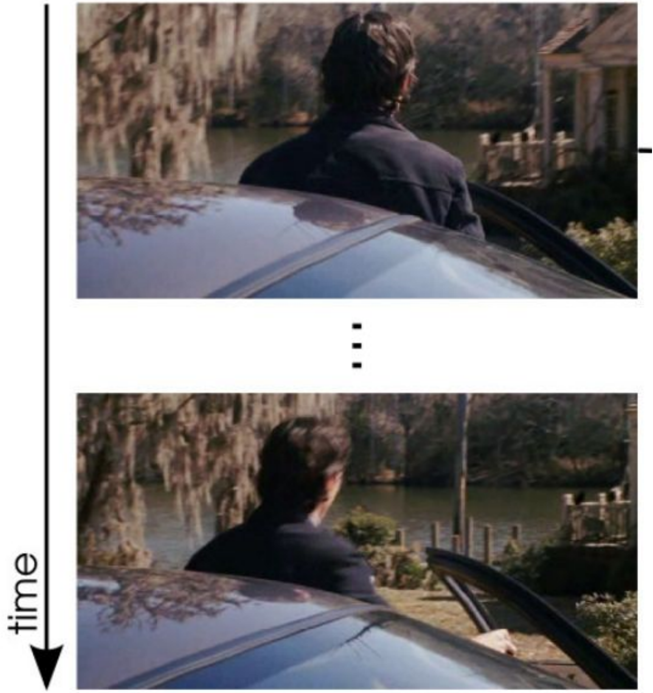
- if not connected to a prev track, start a new one
- when something moves
  - the track it moves to ends
  - the track it moves from replaces it
  - new track starts where it moves from



Dense Trajectories

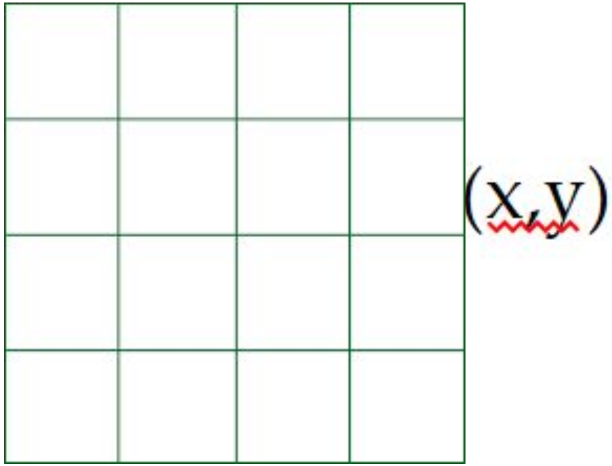
- **if not connected to a prev track, start a new one**
  - the length of a trajectory is limited to L frames
    - when trajectory > L (length), it’s removed
    - because trajectories drift

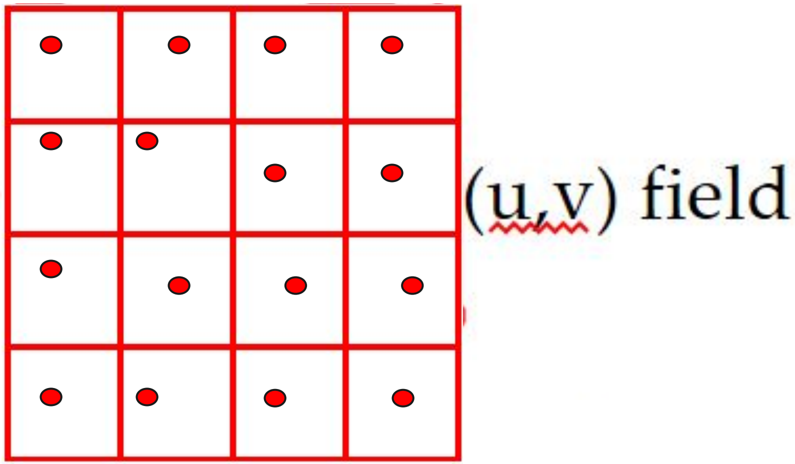




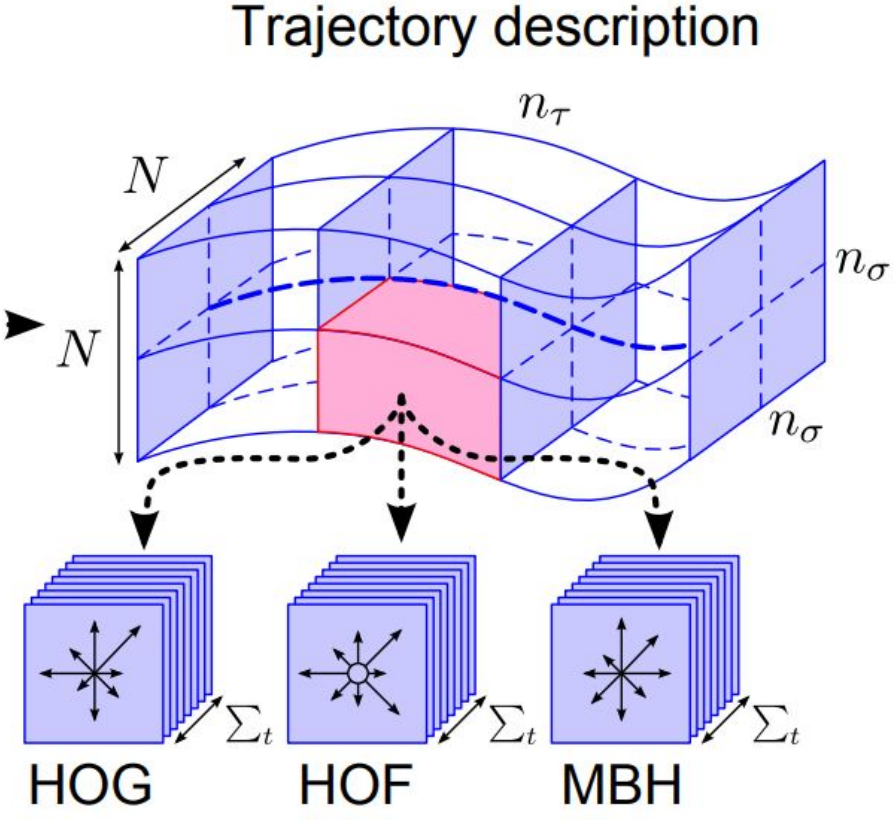
Dense Trajectories

- update each point in box with median of all optical flow (u,v) vectors of that box





Trajectories → Descriptors



Descriptors

- Trajectory
    - Displacement vector $\underline{\Delta P_t,}$
- HOG
- HOF
- MBH

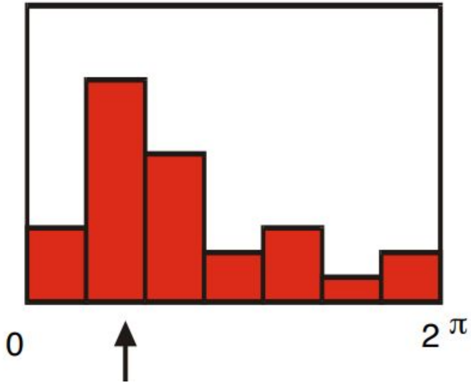
$$S' = \frac{(\Delta P_t, \ldots, \Delta P_{t+L-1})}{\sum_{j=t}^{t+L-1} ||\Delta P_j||}$$

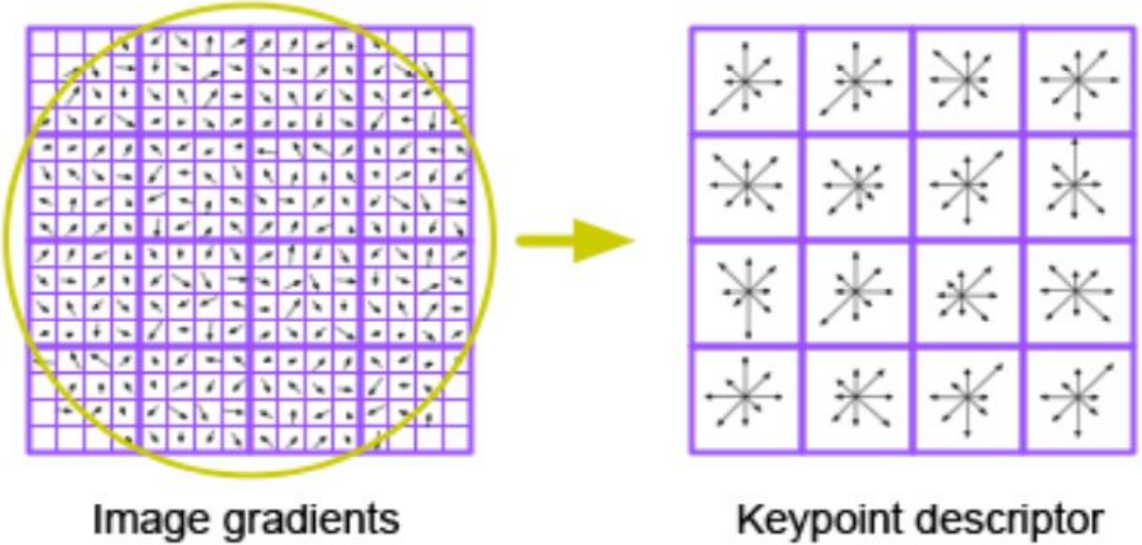
Descriptors

- HOG
  - Apply x and y derivative filters → (Ix, Iy) → (magnitude, direction) → histogram them over NxN pixels
- HOF
- MBH

Descriptors

- HOG:

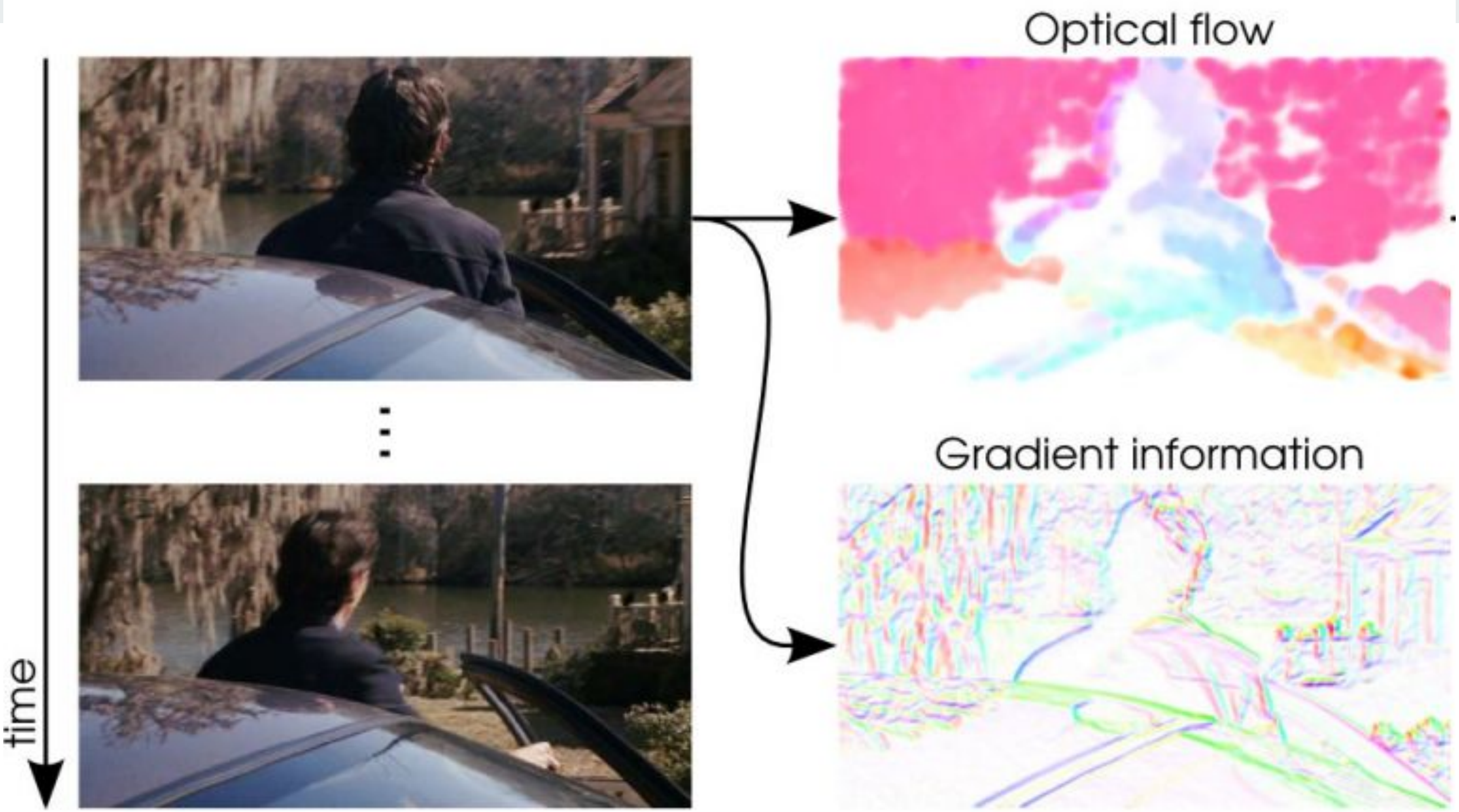




David G. Lowe. "Distinctive image features from scale-invariant keypoints.” *IJCV* 60 (2), pp. 91-110, 2004.

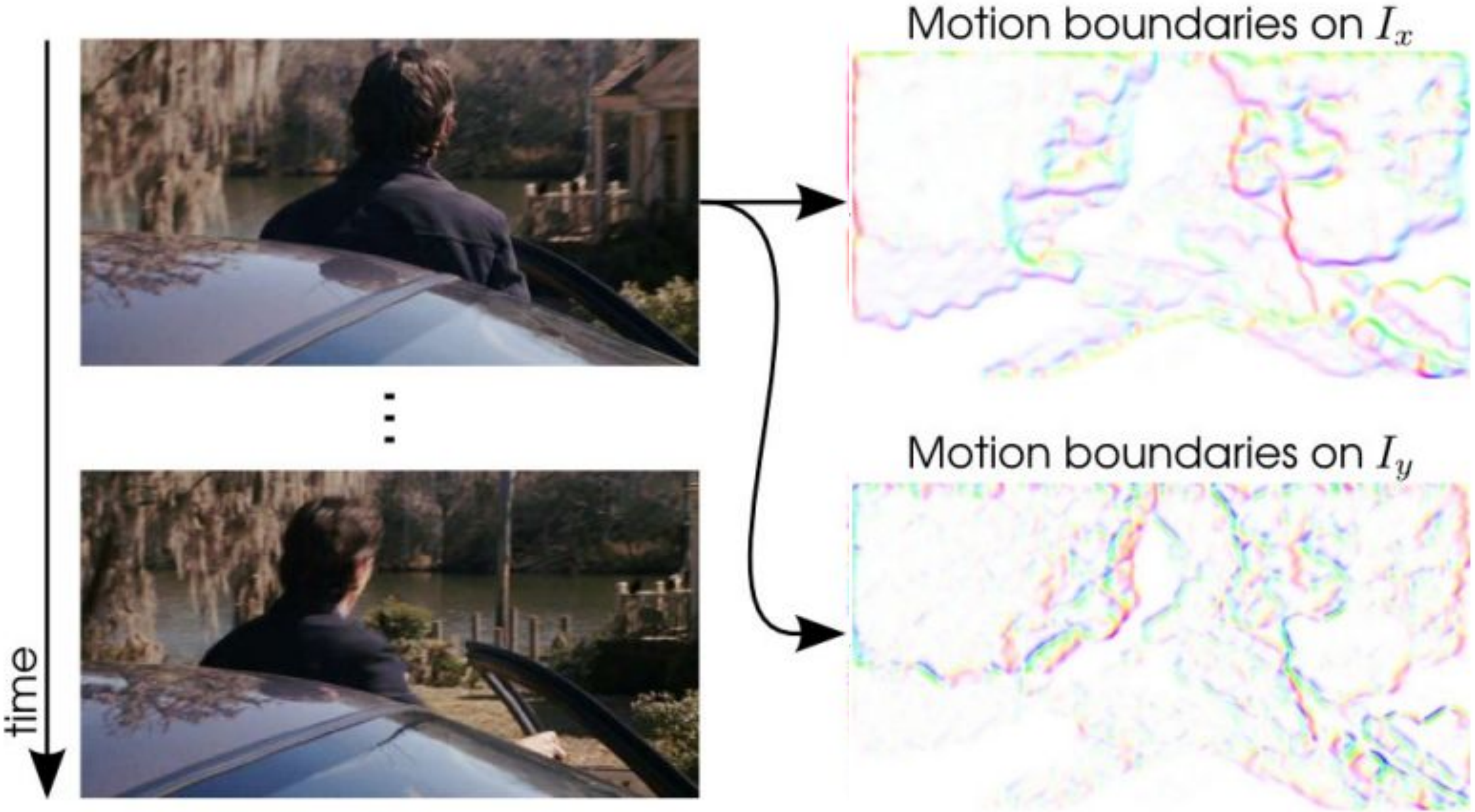
Descriptors

- HOG
- HOF
  - **Same as HOG but instead of (Ix, Iy), use optical flow (u, v) = (It/Ix, It/Iy) → (mag, dir) → histogram over NxN pixels**
- MBH


Optical flow
Gradient information
time

Descriptors

- HOG
- HOF
- **MBH**
  - **Same as HOF but histogram is weighted by the magnitude and expressed as (Ix, Iy)**


time
Motion boundaries on I_x
Motion boundaries on I_y

Specifics of Experimental Setup

- Sampling step size of $W = 5$ is dense enough to give good results
- Used 8 spatial scales spaced by $1/\sqrt{2}$
- Experimentally, trajectory length $L = 15$ frames
- Voxel is $n\sigma \times n\sigma \times n\tau$ where $n\sigma = 2$, $n\tau = 3$

Specifics of Evaluation Setup

- take 100,000 random samples of hog descriptors from all the training videos
- k-means cluster them into 4,000 words
- map any new descriptor to those words
- Classify using a non-linear SVM with a chi-squared kernel

4 Important Datasets

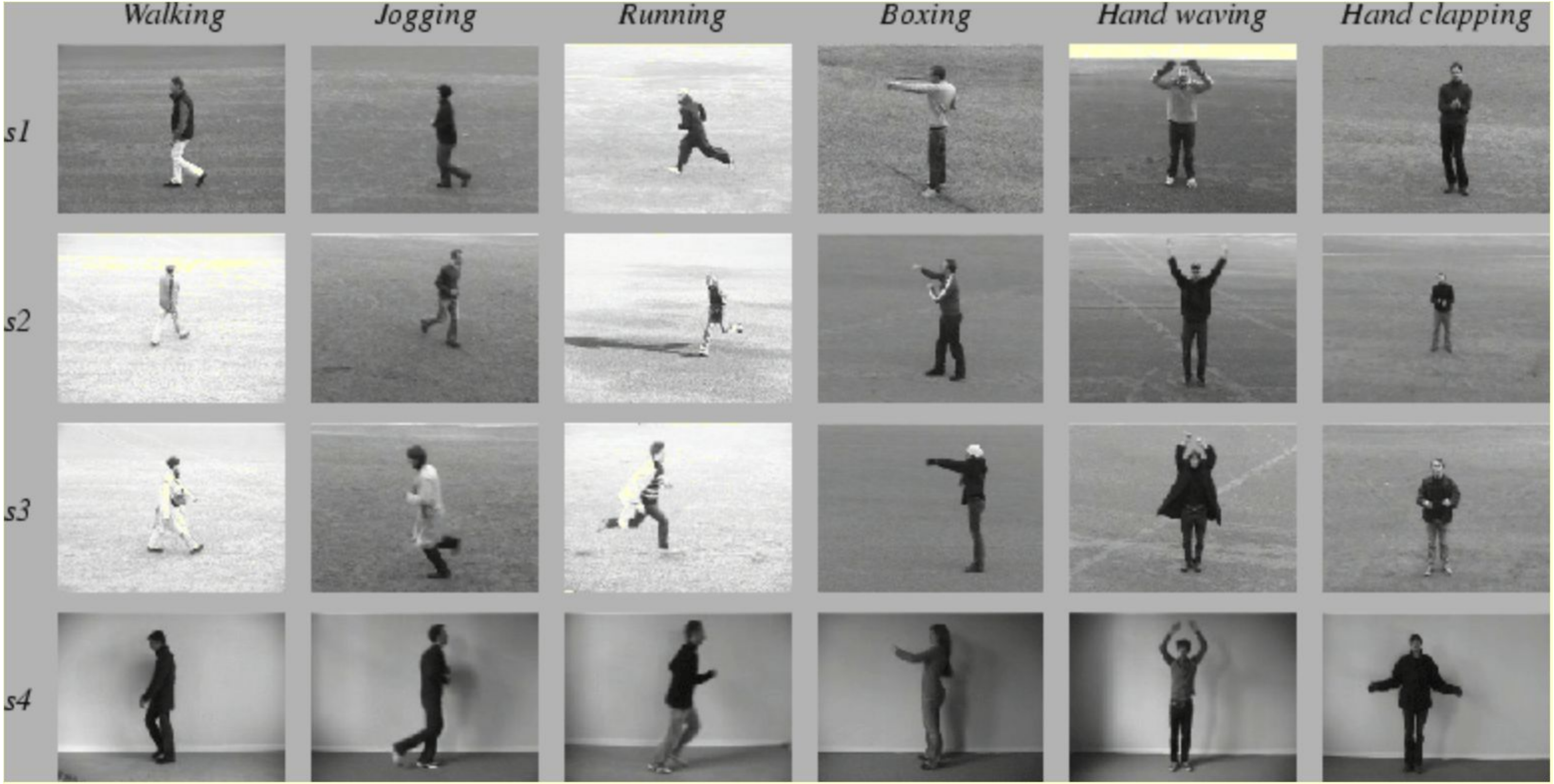
- KTH Human Actions (2004)
- Hollywood-2 (2009)
- HMDB-51: Human Motion DataBase (2011)
- UCF 101 (2012)

KTH Human Actions (2004)

- ‘Simple’/’Controlled’ clips intentionally captured
- Total **2391** clips
- **25** FPS; Avg length of **4** sec (~100 frame clips)
- **160x120** spatial resolution
- Homogeneous background
- Static camera

KTH Human Actions (2004)

- **6** action classes; **4** scenarios; **25** actors;
- Homogeneous background; static camera



Hollywood-2 (2009)

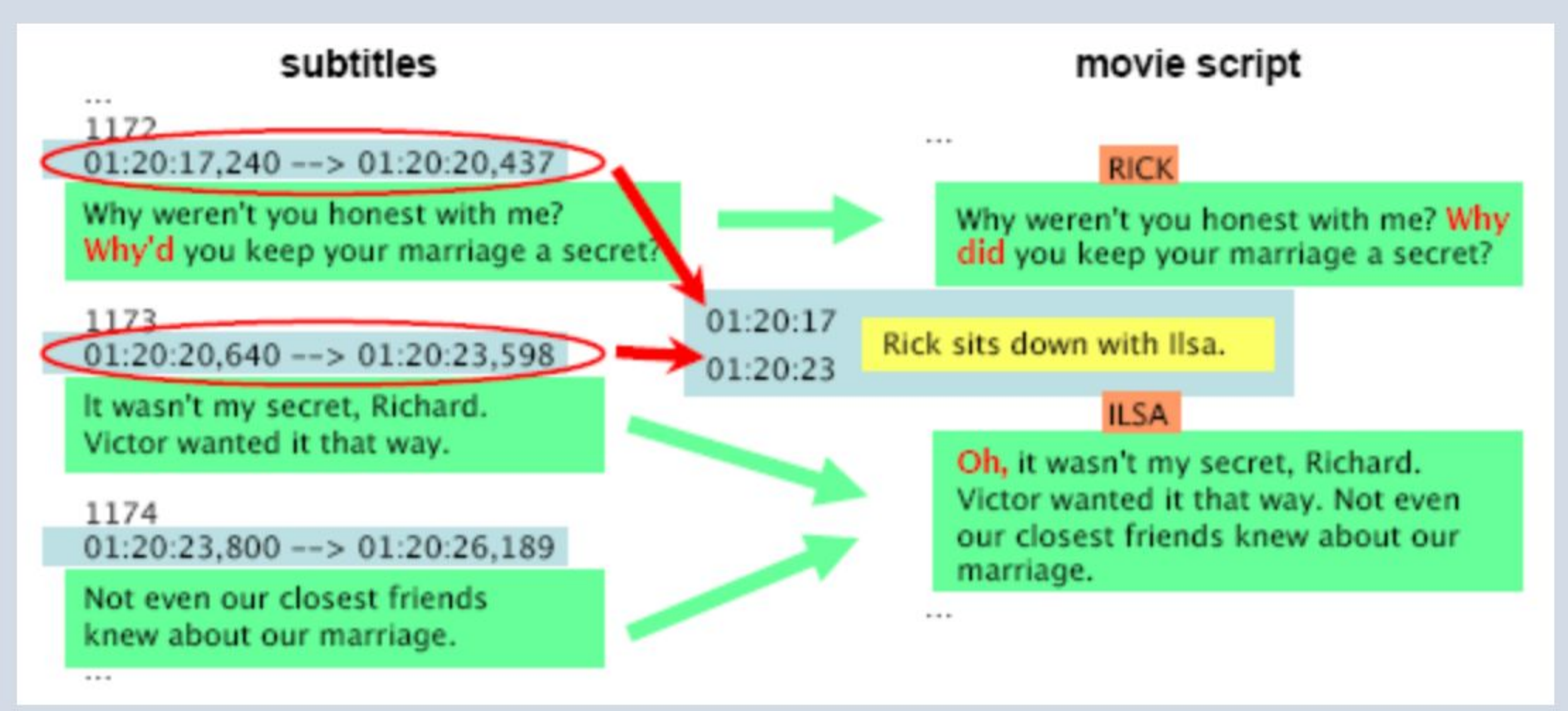
- **12** action classes; **10** scene classes; total **3669** clips
- ~**20.1 hours** of video in total
- Clips from **69 Hollywood Movies** (**different movies** for test & train)
- **Automatic action annotation (!)**
- **Manual verification** afterwards to clean-up

Hollywood-2 (2009)

- **Scripts** describe with **scenes**, **characters**, **transcribed dialogs** and **human action** (free online websites..)
- **Subtitles** have **time** information but only precise speech

- Align **speech sections** between subtitles and scripts
- **Transfer time information** to scene descriptions in scripts

Hollywood-2 (2009)



subtitles

...

1172
01:20:17,240 --> 01:20:20,437
Why weren't you honest with me?
Why'd you keep your marriage a secret?

1173
01:20:20,640 --> 01:20:23,598
It wasn't my secret, Richard.
Victor wanted it that way.

1174
01:20:23,800 --> 01:20:26,189
Not even our closest friends
knew about our marriage.

...

movie script

...

RICK

Why weren't you honest with me? Why did you keep your marriage a secret?

01:20:17
01:20:23
Rick sits down with Ilsa.

ILSA

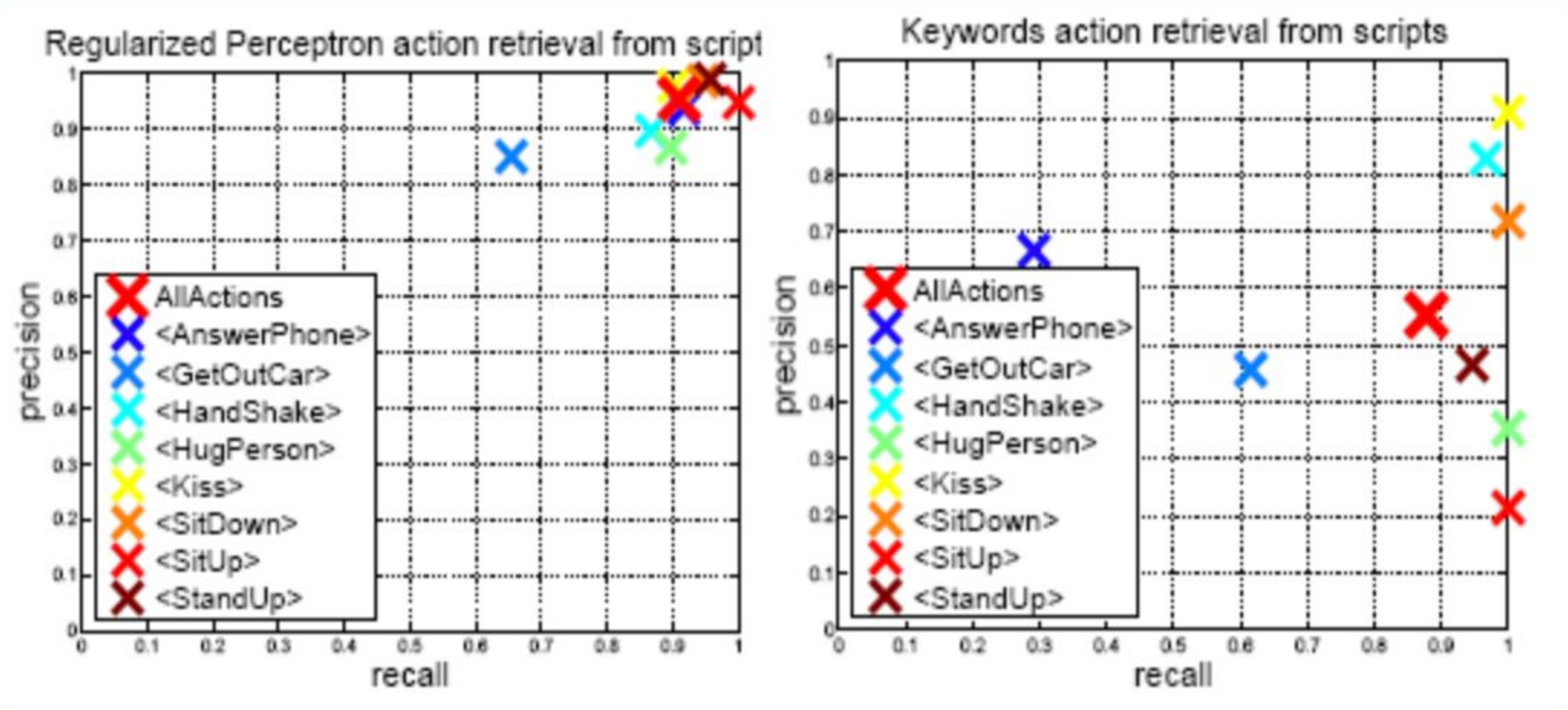
Oh, it wasn't my secret, Richard. Victor wanted it that way. Not even our closest friends knew about our marriage.

...

A “Text” Action Classifier

- Train a **Regularized Perceptron text classifier** for each action class
- Assign action labels to scene descriptions
- Does much better than hand-tuned regular-expression matching

Hollywood-2 (2009)



Hollywood-2 (2009)

- Sample video clips can contain multiple actions (probably true of other datasets too)
- Also gave conditional probability estimates: **p(scene|action)** and **p(action|scene)** using the movie scripts for **clips not cleaned**

Actions

	Training subset (clean)	Training subset (automatic)	Test subset (clean)
AnswerPhone	66	59	64
DriveCar	85	90	102
Eat	40	44	33
FightPerson	54	33	70
GetOutCar	51	40	57
HandShake	32	38	45
HugPerson	64	27	66
Kiss	114	125	103
Run	135	187	141
SitDown	104	87	108
SitUp	24	26	37
StandUp	132	133	146
All Samples	**823**	**810**	**884**

Scenes

	Training subset (automatic)	Test subset (clean)
EXT-House	81	140
EXT-Road	81	114
INT-Bedroom	67	69
INT-Car	44	68
INT-Hotel	59	37
INT-Kitchen	38	24
INT-LivingRoom	30	51
INT-Office	114	110
INT-Restaurant	44	36
INT-Shop	47	28
All Samples	**570**	**582**

HMDB (2011)

- **7000** manually annotated clips from **YouTube** & **Movies**
- **51** action classes (**>= 100** clips each)
- **90+%** accuracy on existing popular datasets (KTH, Weizmann etc)

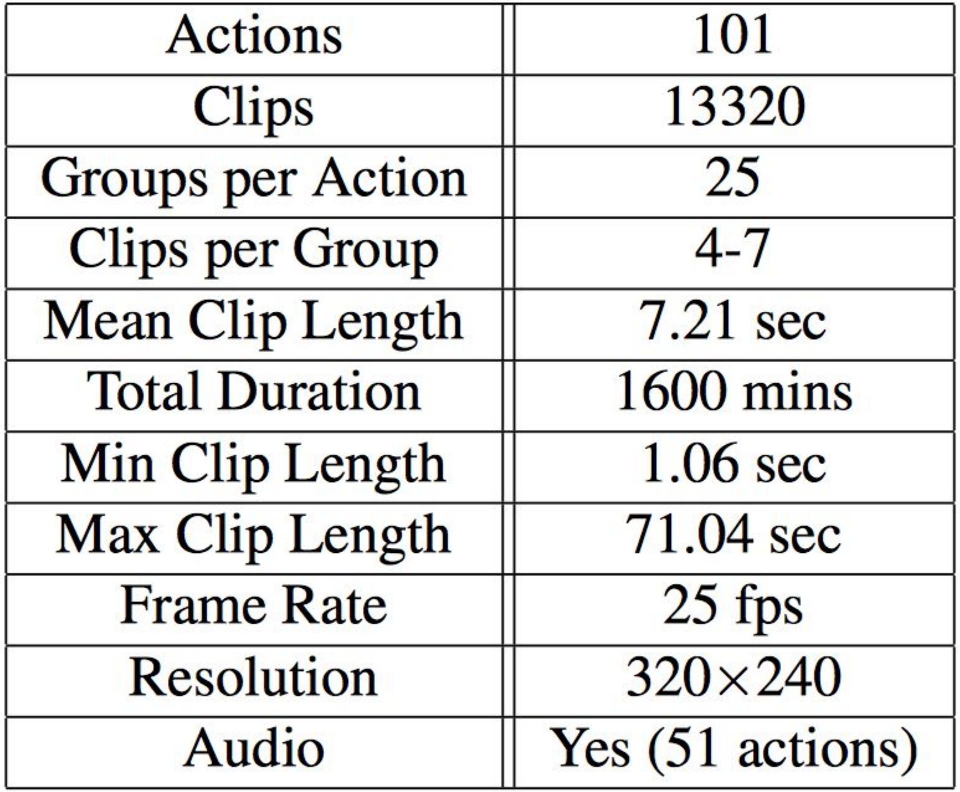
- Interesting experiment to show that **HMDB's** action categories mainly differ in **motion** rather than **static poses**
- Contrary to **UCF-50** & **Hollywood2:** "solvable" with static information alone

- **Shown on Left**:
  i) Hand-waving
  ii) Drinking
  iii) Sword Fighting
  iv) Diving
  v) Running
  vi) Kicking
- **Large variation** in camera viewpoint/motion**,** cluttered background**,** position/scale/appearance of actors

UCF 101 (2012)

- “Realistic” action videos taken from **YouTube**
- Extension of earlier **UCF-50**
- Examples: …*Apply Eye Makeup, Archery, Baby Crawling, Blowing Candles, Body Weight Squats, Boxing Punching Bag, Hammering, Handstand Push-ups, Handstand Walking, Walking with a dog, Wall Push-ups…*
- ~**Twice** as big as **UCF-50**, **HMDB-51**
- Authors’ claim: “**Most challenging data set to date**”;
  Largest diversity in actions, variations in camera motion, object appearance/pose/scale/viewpoint, cluttered background, illumination conditions..
- **Authors’ Baseline result** (w/ standard BOW approach)**: 43.90%**

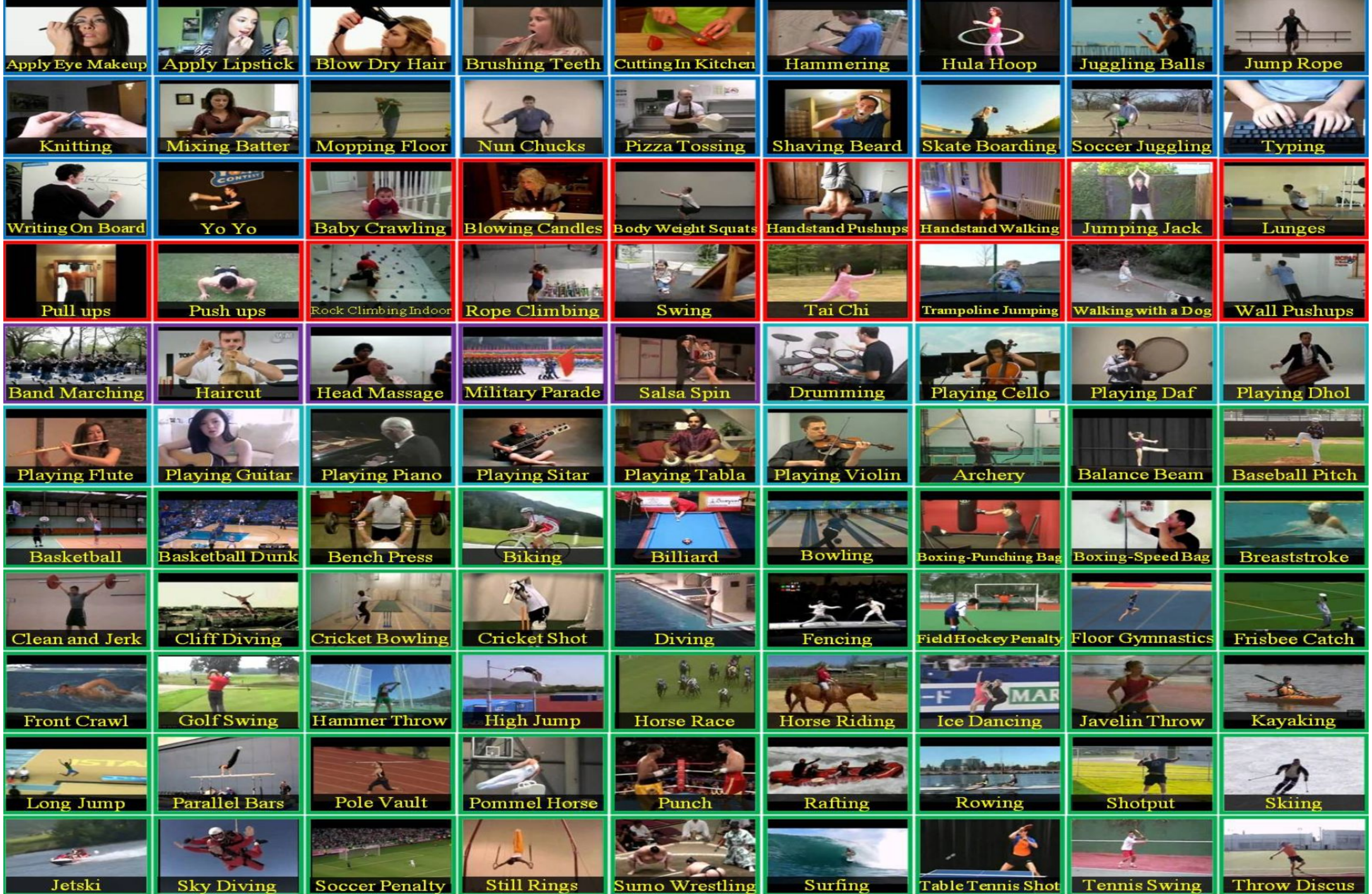
UCF 101 (2012)



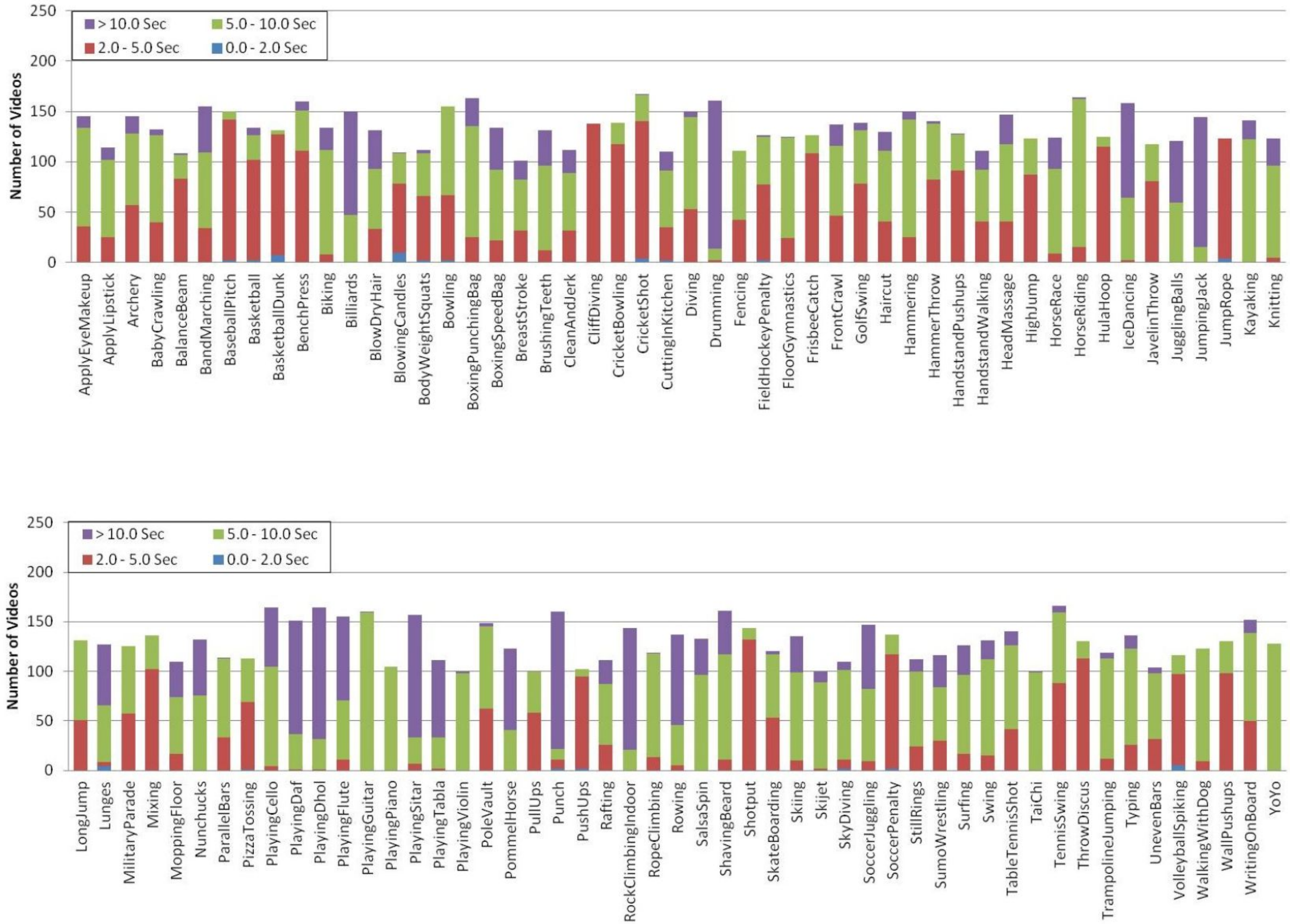
Actions	101
Clips	13320
Groups per Action	25
Clips per Group	4-7
Mean Clip Length	7.21 sec
Total Duration	1600 mins
Min Clip Length	1.06 sec
Max Clip Length	71.04 sec
Frame Rate	25 fps
Resolution	320×240
Audio	Yes (51 actions)

Table 1. Summary of Characteristics of UCF101

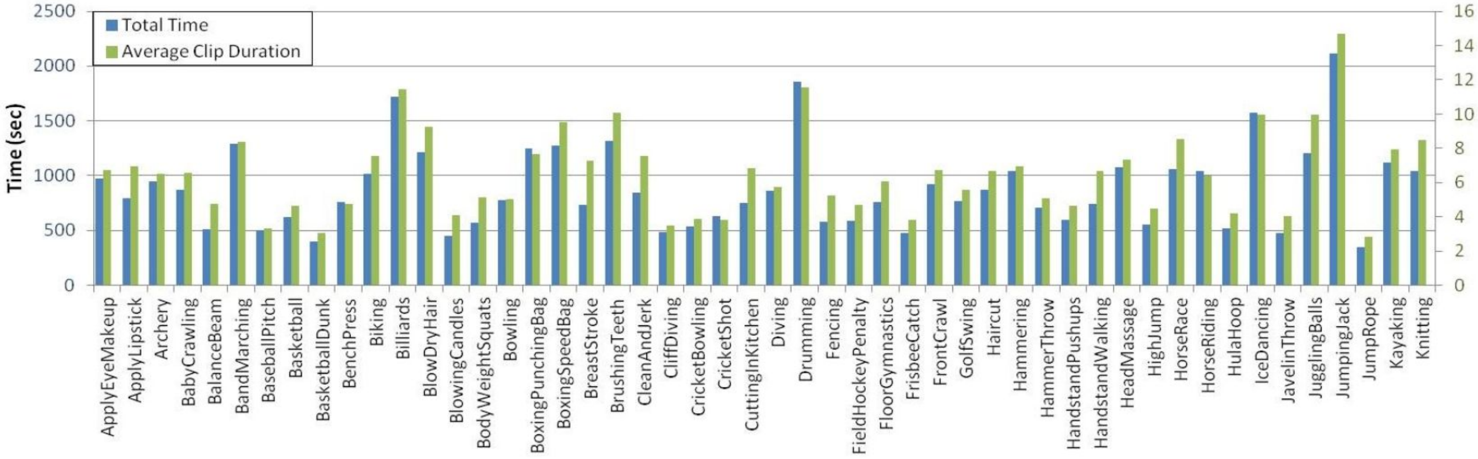
- **5** broad **categories** of actions (**101**):
  i) Human-Object Interaction
  ii) Body-Motion Only
  iii) Human- Human Interaction
  iv) Playing Musical Instruments
  v) Sports
- **25 Groups (4-7 videos):** clips with commonalities (similar background, viewpoint etc)

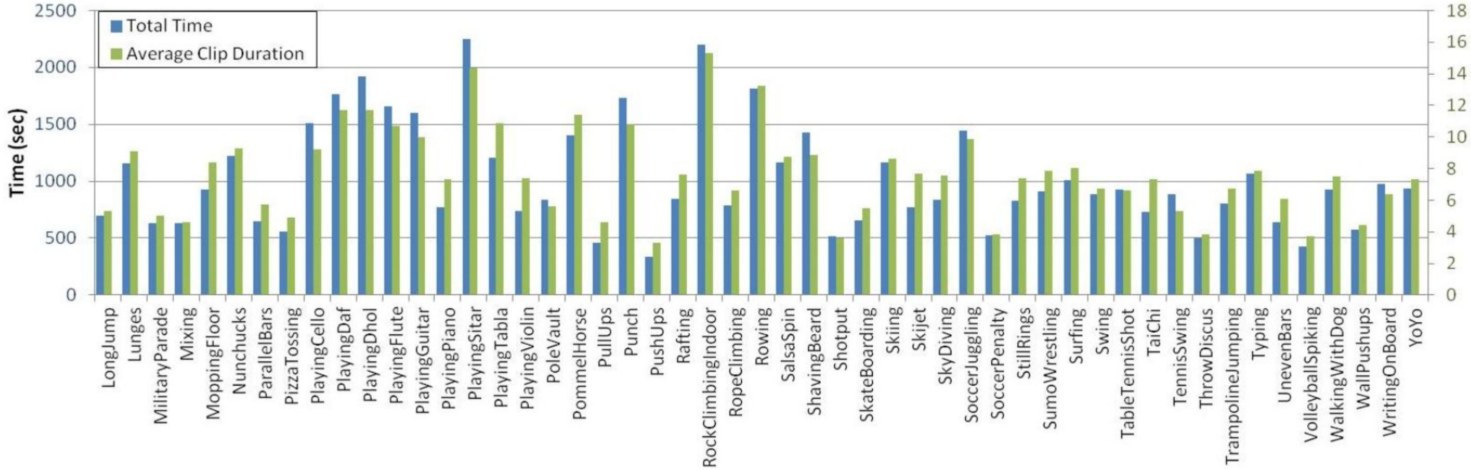

Apply Eye Makeup
Apply Lipstick
Blow Dry Hair
Brushing Teeth
Cutting In Kitchen
Hammering
Hula Hoop
Juggling Balls
Jump Rope
Knitting
Mixing Batter
Mopping Floor
Nun Chucks
Pizza Tossing
Shaving Beard
Skate Boarding
Soccer Juggling
Typing
Writing On Board
Yo Yo
Baby Crawling
Blowing Candles
Body Weight Squats
Handstand Pushups
Handstand Walking
Jumping Jack
Lunges
Pull ups
Push ups
Rock Climbing Indoor
Rope Climbing
Swing
Tai Chi
Trampoline Jumping
Walking with a Dog
Wall Pushups
Band Marching
Haircut
Head Massage
Military Parade
Salsa Spin
Drumming
Playing Cello
Playing Daf
Playing Dhol
Playing Flute
Playing Guitar
Playing Piano
Playing Sitar
Playing Tabla
Playing Violin
Archery
Balance Beam
Baseball Pitch
Basketball
Basketball Dunk
Bench Press
Biking
Billiard
Bowling
Boxing-Punching Bag
Boxing-Speed Bag
Breaststroke
Clean and Jerk
Cliff Diving
Cricket Bowling
Cricket Shot
Diving
Fencing
Field Hockey Penalty
Floor Gymnastics
Frisbee Catch
Front Crawl
Golf Swing
Hammer Throw
High Jump
Horse Race
Horse Riding
Ice Dancing
Javelin Throw
Kayaking
Long Jump
Parallel Bars
Pole Vault
Pommel Horse
Punch
Rafting
Rowing
Shotput
Skiing
Jetski
Sky Diving
Soccer Penalty
Still Rings
Sumo Wrestling
Surfing
Table Tennis Shot
Tennis Swing
Throw Discus

Number Of Videos



Clip Lengths

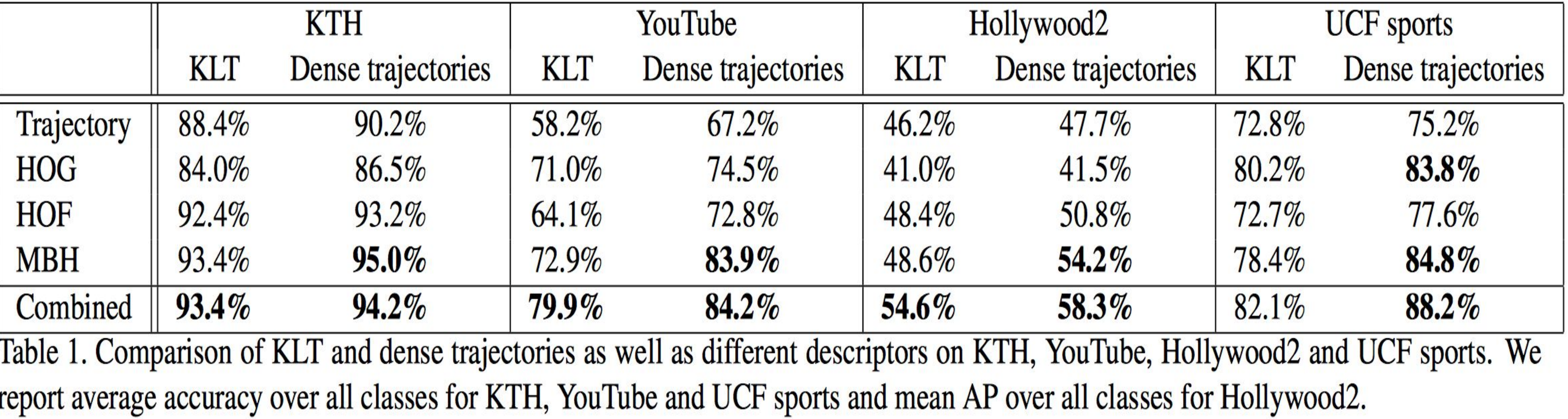




Evaluation

Action Recognition by Dense Trajectories *by Wang, Klaser, Schmid, Cheng-Lin CVPR'11*

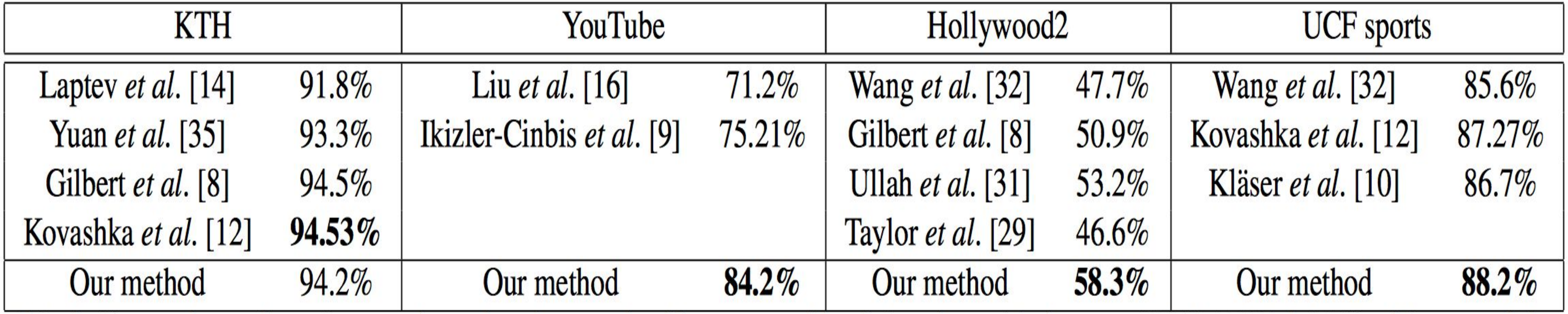
Results on four Datasets



	KTH		YouTube		Hollywood2		UCF sports	
	KLT	Dense trajectories	KLT	Dense trajectories	KLT	Dense trajectories	KLT	Dense trajectories
Trajectory	88.4%	90.2%	58.2%	67.2%	46.2%	47.7%	72.8%	75.2%
HOG	84.0%	86.5%	71.0%	74.5%	41.0%	41.5%	80.2%	**83.8%**
HOF	92.4%	93.2%	64.1%	72.8%	48.4%	50.8%	72.7%	77.6%
MBH	93.4%	**95.0%**	72.9%	**83.9%**	48.6%	**54.2%**	78.4%	**84.8%**
Combined	**93.4%**	**94.2%**	**79.9%**	**84.2%**	**54.6%**	**58.3%**	82.1%	**88.2%**

Table 1. Comparison of KLT and dense trajectories as well as different descriptors on KTH, YouTube, Hollywood2 and UCF sports. We report average accuracy over all classes for KTH, YouTube and UCF sports and mean AP over all classes for Hollywood2.

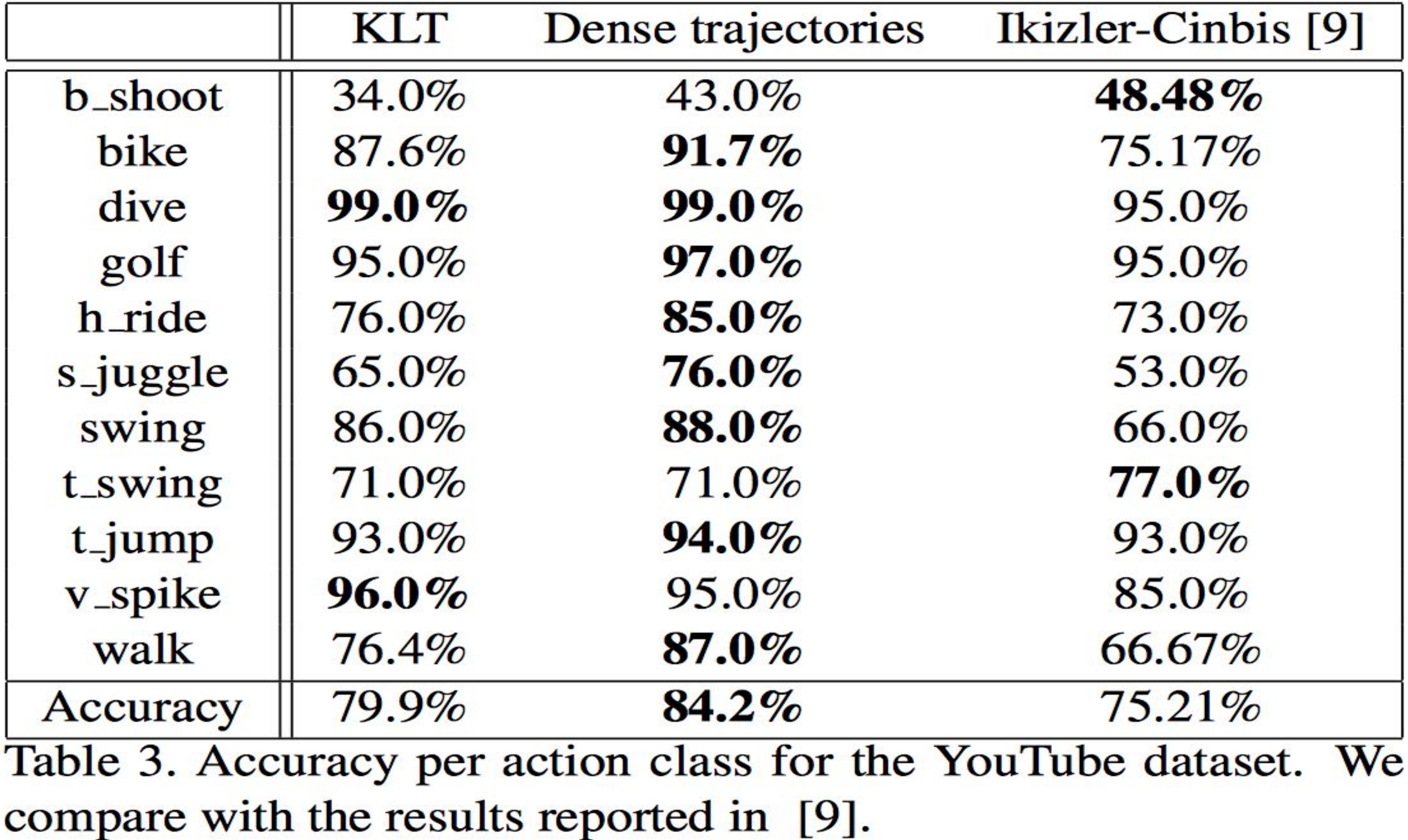
Comparing to the **state-of-the-art**



KTH		YouTube		Hollywood2		UCF sports	
Laptev *et al.* [14]	91.8%	Liu *et al.* [16]	71.2%	Wang *et al.* [32]	47.7%	Wang *et al.* [32]	85.6%
Yuan *et al.* [35]	93.3%	Ikizler-Cinbis *et al.* [9]	75.21%	Gilbert *et al.* [8]	50.9%	Kovashka *et al.* [12]	87.27%
Gilbert *et al.* [8]	94.5%			Ullah *et al.* [31]	53.2%	Kläser *et al.* [10]	86.7%
Kovashka *et al.* [12]	**94.53%**			Taylor *et al.* [29]	46.6%		
Our method	94.2%	Our method	**84.2%**	Our method	**58.3%**	Our method	**88.2%**

Table 2. Comparison of our dense trajectories characterized by our combined descriptor (Trajectory+HOG+HOF+MBH) with state-of-the-art methods in the literature.

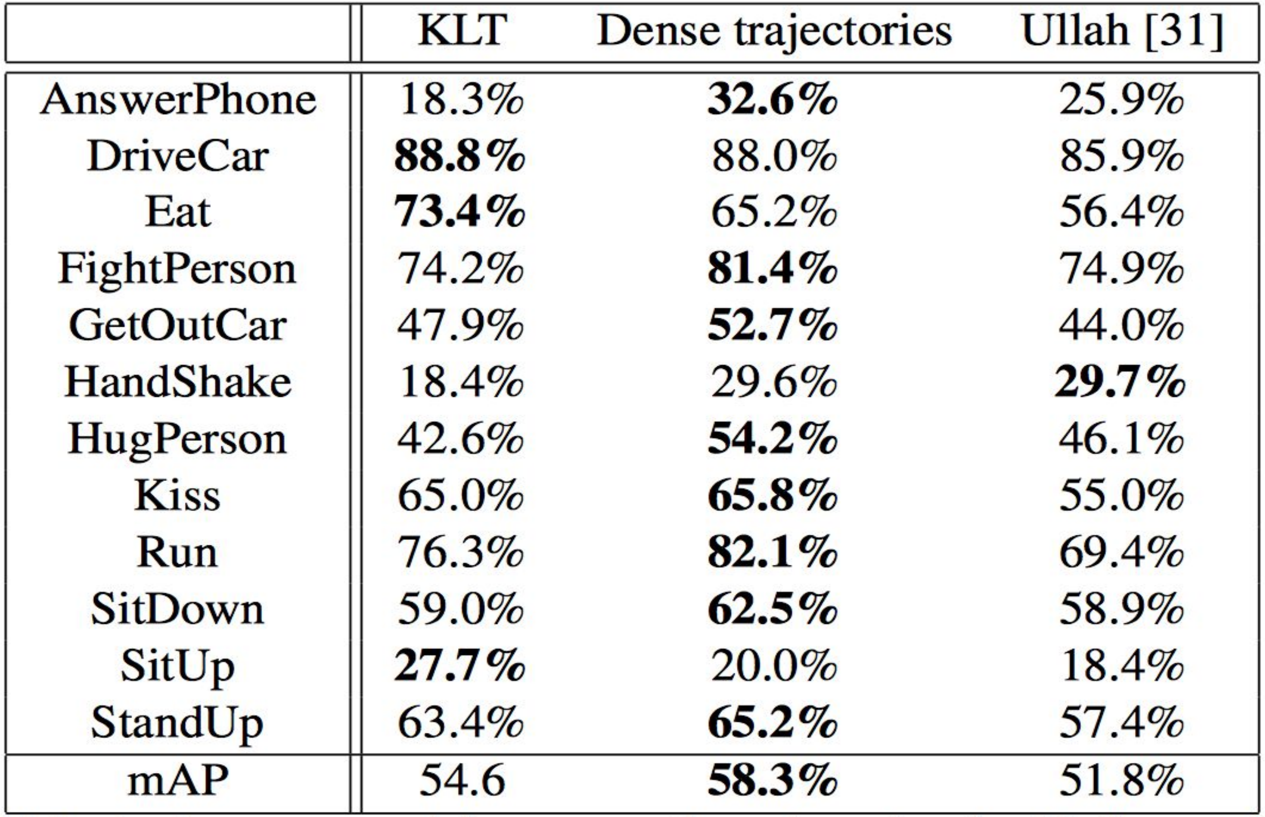
Per-class **accuracy** analysis on **YouTube**



	KLT	Dense trajectories	Ikizler-Cinbis [9]
b_shoot	34.0%	43.0%	**48.48%**
bike	87.6%	**91.7%**	75.17%
dive	**99.0%**	**99.0%**	95.0%
golf	95.0%	**97.0%**	95.0%
h_ride	76.0%	**85.0%**	73.0%
s_juggle	65.0%	**76.0%**	53.0%
swing	86.0%	**88.0%**	66.0%
t_swing	71.0%	71.0%	**77.0%**
t_jump	93.0%	**94.0%**	93.0%
v_spike	**96.0%**	95.0%	85.0%
walk	76.4%	**87.0%**	66.67%
Accuracy	79.9%	**84.2%**	75.21%

Table 3. Accuracy per action class for the YouTube dataset. We compare with the results reported in [9].

Per-class **AP** analysis on **Hollywood2**



	KLT	Dense trajectories	Ullah [31]
AnswerPhone	18.3%	**32.6%**	25.9%
DriveCar	**88.8%**	88.0%	85.9%
Eat	**73.4%**	65.2%	56.4%
FightPerson	74.2%	**81.4%**	74.9%
GetOutCar	47.9%	**52.7%**	44.0%
HandShake	18.4%	29.6%	**29.7%**
HugPerson	42.6%	**54.2%**	46.1%
Kiss	65.0%	**65.8%**	55.0%
Run	76.3%	**82.1%**	69.4%
SitDown	59.0%	**62.5%**	58.9%
SitUp	**27.7%**	20.0%	18.4%
StandUp	63.4%	**65.2%**	57.4%
mAP	54.6	**58.3%**	51.8%

Table 4. Average precision per action class for the Hollywood2 dataset. We compare with the results reported in [31].

Varying **hyper-parameters** of the System

- **L** (Trajectory Length)
- **W** (Step Size)
- **N** (Neighborhood Size)
- **n_sigma** * **n_sigma** * **n_tau** (Grid Structure)

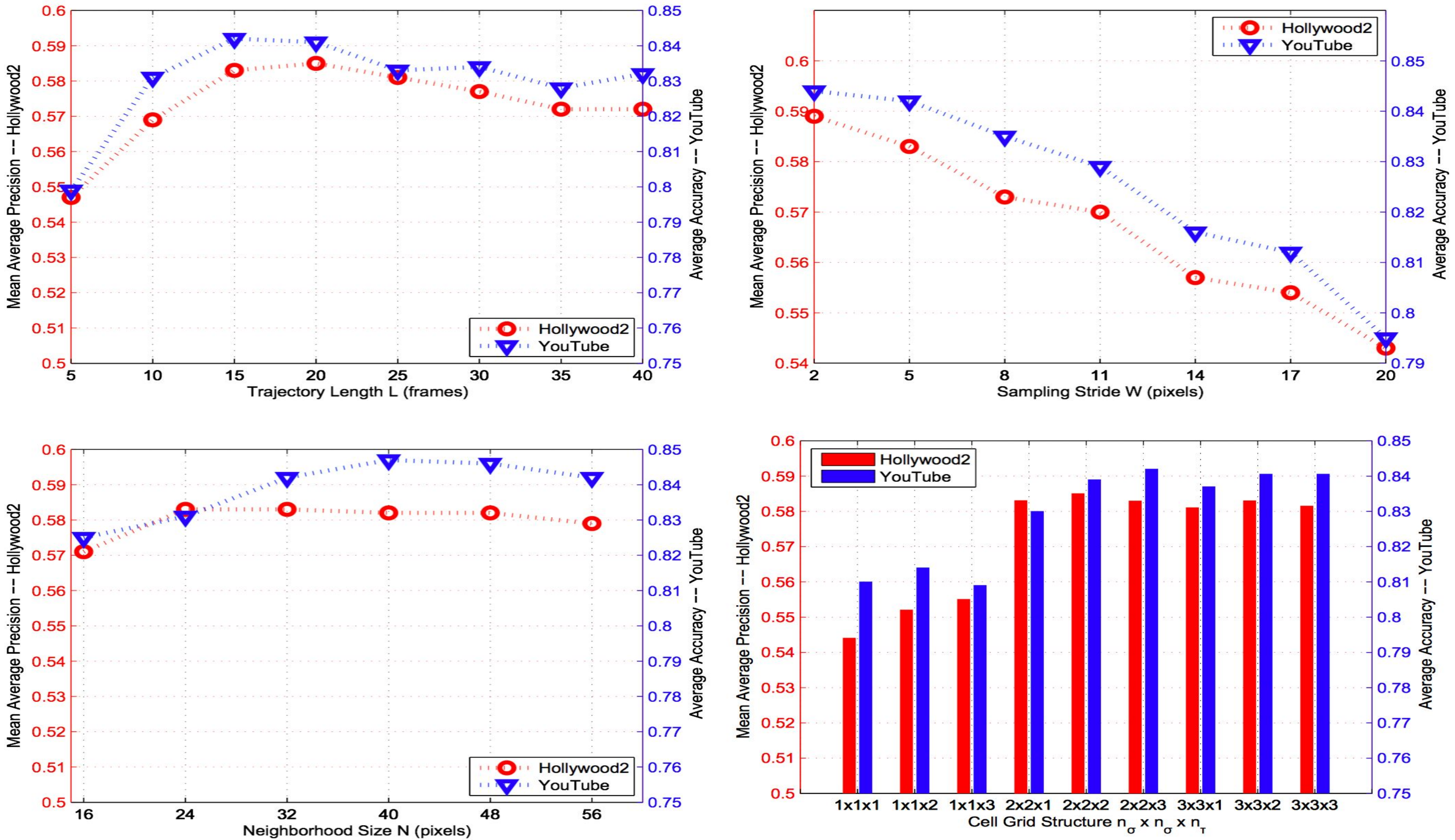


Figure 5. Results for different parameter settings on the Hollywood2 and YouTube datasets.

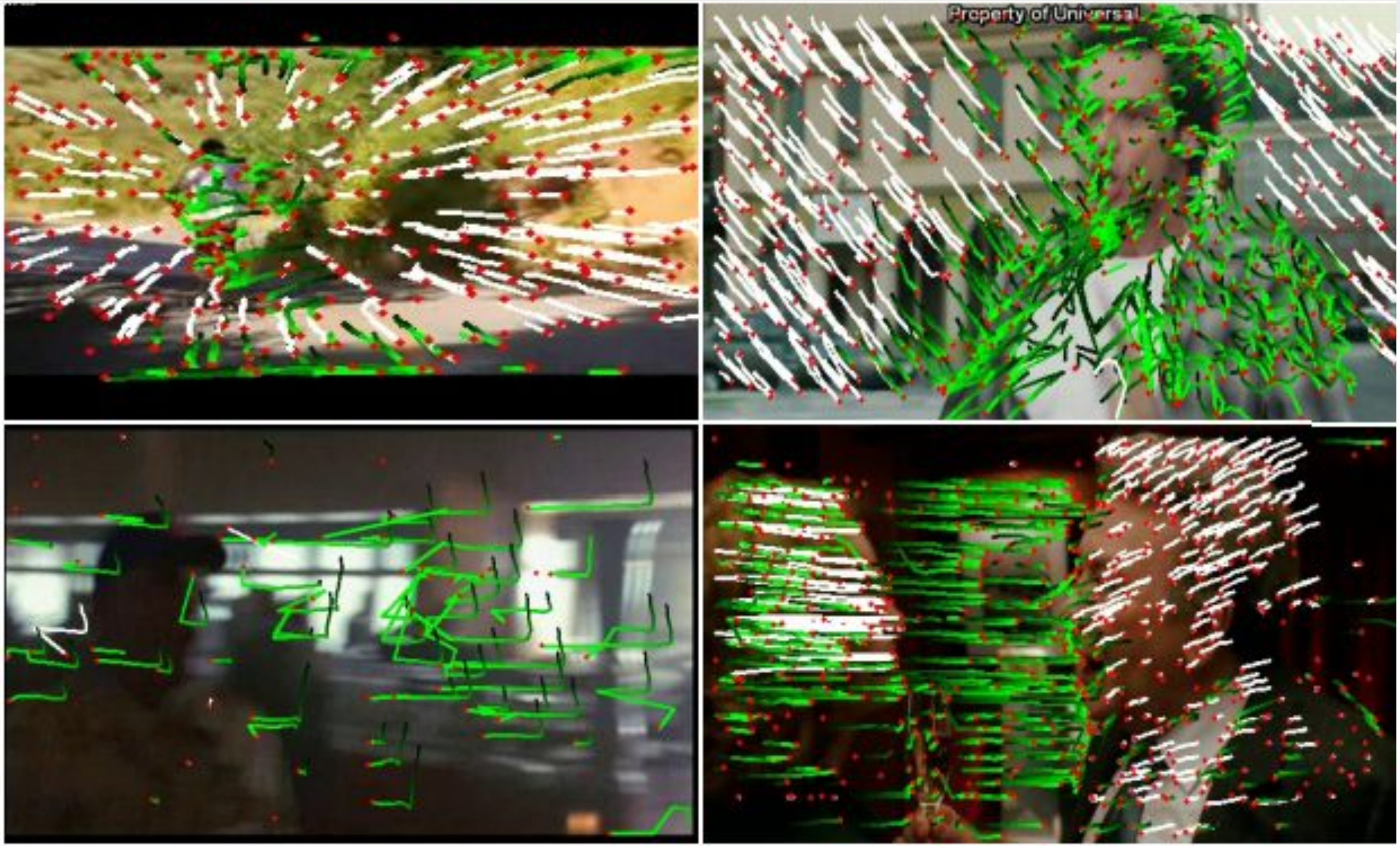
Improvements made by IDT:

1. Uses 2 types of features:
    - SURF focuses on blob-type structures,
    - Harris Corner Detector fires on corners edges
2. Remove majority trajectories
    - RANSAC on optical flow,SURF, & Harris
    - Correct for **majority** homography
    - threshold magnitude of the (u, v)

Imp

1. U

2. E



More Improvements made by IDT:

3. Human detector bounding box
   - mask to remove feature matches inside for when homography
4. Fisher Vector > Bag of Features
   - Uses PCA, Gaussian Mixture model, then classifies by **Linear SVM**

Thoughts on the paper (!!)

- ...

;