Video Understanding in the Wild with Incomplete Supervision

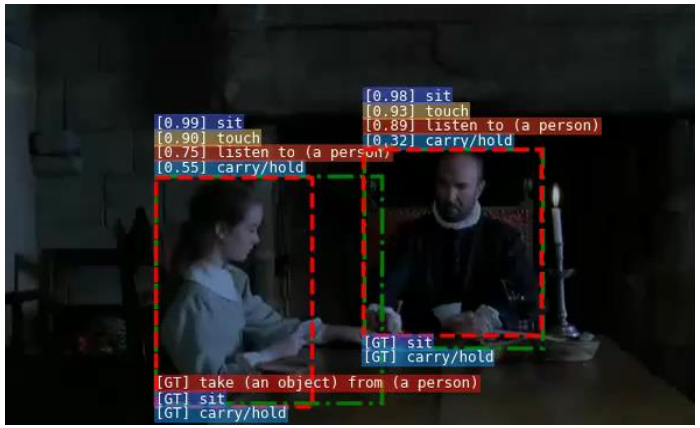
Anurag Arnab

Google Research

Video Understanding

Human Pose Estimation

Video Segmentation



Action recognition

Exploiting Temporal Context for 3D Pose Estimation in the Wild

Anurag Arnab, Carl Doersch,
Andrew Zisserman

Google Research

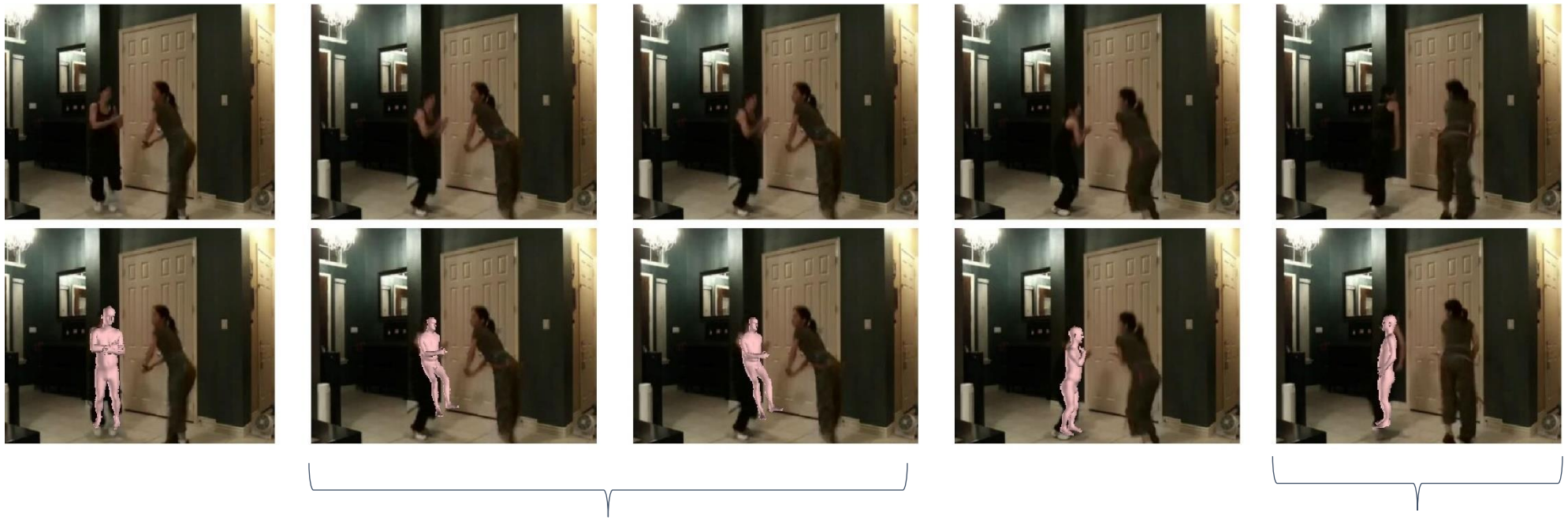
Introduction

- Monocular 3D human pose estimation is an inherently ill-posed problem
- Metric ground-truth for real-world data is prohibitively difficult to collect
- Common datasets are from motion capture in controlled labs
- Models trained on these datasets generalise poorly to “in the wild”

Human 3.6M (mocap dataset)

“In-the-wild” data

Generalising to the real-world



[1] Kanazawa *et al.* End-to-end recovery of human shape and pose. *CVPR* 2018.

Temporal Consistency

- Temporal dimension of ordinary video encodes valuable information
- Multiple views of person observed
  - Body shape and bone lengths remain constant
  - Joint positions in both 2D and 3D vary slowly

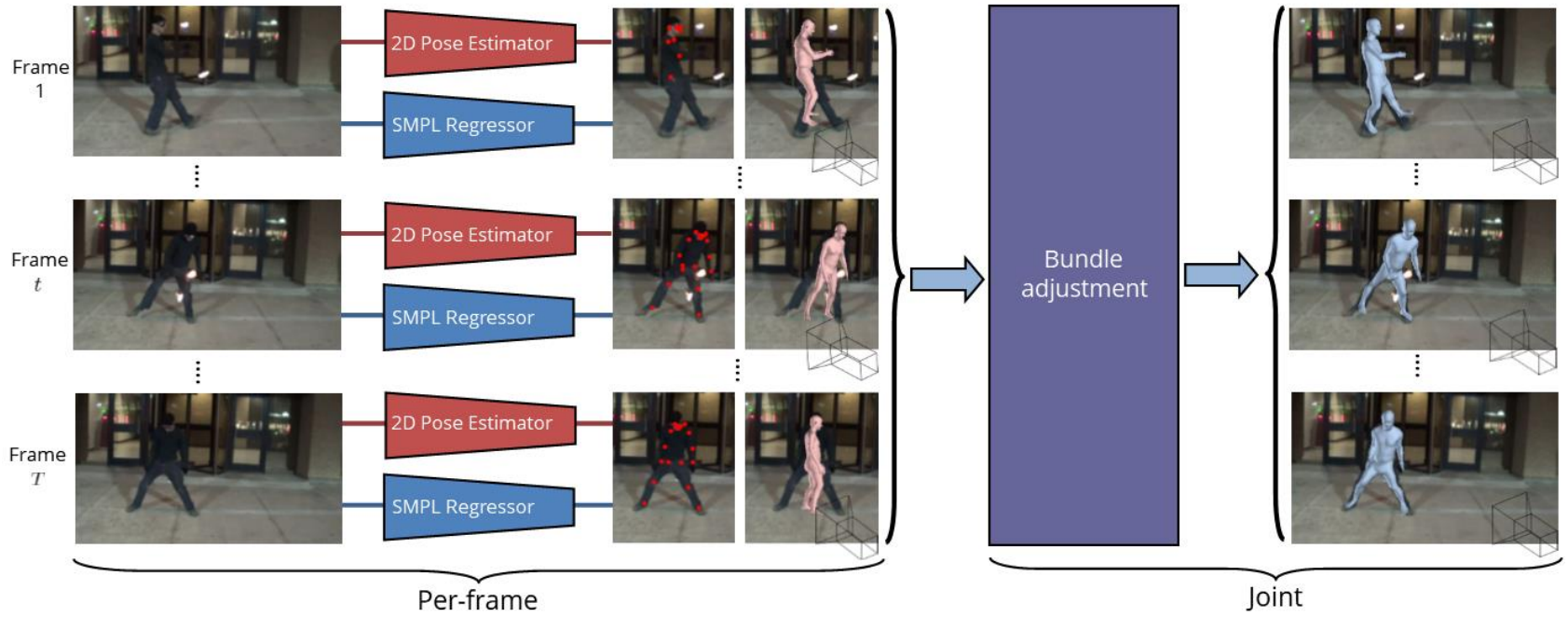
Our approach

- Propose a form of bundle adjustment
  - Take into account temporal information and multi-view geometry of the video
- Apply our method to about 107 000 YouTube videos in the Kinetics dataset.
- Automatically create a new “in-the-wild” dataset from this
- Improve performance of per-frame model using this dataset.

Bundle Adjustment

- Objective function consists of reprojection, temporal and prior terms

$$E(\beta, \theta, \mathbf{\Omega}) = E_R(\beta, \theta, \mathbf{\Omega}) + E_T(\beta, \theta, \mathbf{\Omega}) + E_P(\theta, \beta)$$

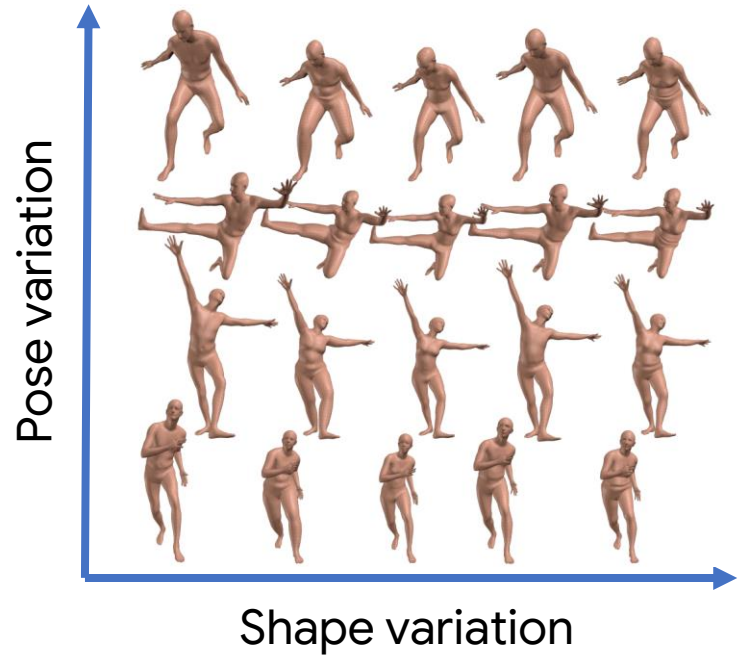


Bundle Adjustment

- Objective function consists of reprojection, temporal and prior terms

$$E(\beta, \theta, \mathbf{\Omega}) = E_R(\beta, \theta, \mathbf{\Omega}) + E_T(\beta, \theta, \mathbf{\Omega}) + E_P(\theta, \beta)$$

- Use the SMPL body model [1] to parameterise the 3D pose
- Pose parameters: $\theta^t \in \mathbb{R}^{23\times 3}$
- Shape parameters: $\beta \in \mathbb{R}^{10}$
- Shape remains constant throughout the video.



Loper et al. SMPL: A Skinned Multi-Person Linear Model. ECCV 2016.

Bundle Adjustment

- Objective function consists of reprojection, temporal and prior terms

$$E(\beta, \theta, \mathbf{\Omega}) = E_R(\beta, \theta, \mathbf{\Omega}) + E_T(\beta, \theta, \mathbf{\Omega}) + E_P(\theta, \beta)$$

- Use the SMPL body model [1] to parameterise the 3D pose
- 3D joints (and mesh vertices) are obtained from the differentiable SMPL function
- 3D joints, $\mathbf{X}^t = \mathrm{SMPL}(\beta, \theta^t)$
- Assume scaled orthographic projection, Π, with camera parameters $\mathbf{\Omega}^t = \{s^t, u^t\}$
- We can project this onto 2D using the camera parameters.
- 2D joints, $\mathbf{x}^t = s^t \Pi(R\mathbf{X}^t) + u^t$.

Reprojection Error

- Encourage 3D joint to project onto predicted 2D keypoints.

$$E_R(\beta, \theta, \mathbf{\Omega}) = \lambda_R \sum_t^T \sum_i^J w_i \rho(\mathbf{x}_i^t - \mathbf{x}_{det,i}^t).$$

- We use 2D human detector of [1].
- Use robust Huber error function
- And weight each reprojection term by the keypoint detector's confidence.

Input keypoints

Predicted joint projection

Predicted 3D mesh

[1] Papanderou *et al.* Towards accurate multi-person pose in the wild. *CVPR* 2017.

Temporal Error

- Encourage smooth motion of predicted:
  - 3D joints, $\mathbf{X}$
  - 2D joint projection, $\mathbf{x}$
  - camera parameters, $\mathbf{\Omega}$

$$E_T(\beta, \theta, \mathbf{\Omega}) = \sum_{t=2}^{T} \sum_{i=1}^{J} \lambda_1 \rho(\mathbf{X}_i^t - \mathbf{X}_i^{t-1}) + \lambda_2 \rho(\mathbf{x}_i^t - \mathbf{x}_i^{t-1}) + \lambda_3 \rho(\mathbf{\Omega}^t - \mathbf{\Omega}^{t-1})$$

3D Prior

- Many 3D poses (some not humanly possible) that project correctly onto 2D and temporally smooth.
- One term encourages solution to stay close to the initialisation, the other the commonly used GMM pose prior [1].

No prior

Prior

[1] Bogo *et al.* Keep it SMPL ... *ECCV* 2016.

3D Prior

- Many 3D poses (some not humanly possible) that project correctly onto 2D and temporally smooth.
- One term encourages solution to stay close to the initialisation, the other the commonly used GMM pose prior [1].

$$E_P(\beta, \theta) = \sum_t^T E_J(\theta^t) + \lambda_I E_I(\theta^t, \beta)$$

$$E_J(\theta) = -\log\left(\sum_i g_i \mathcal{N}\left(\theta^t; \mu_i, \Sigma_i\right)\right)$$

$$E_I(\theta^t, \beta) = \sum_i^J \rho(\mathbf{X}_i^t - \tilde{\mathbf{X}}_i^t) + \lambda_\beta \rho(\beta - \tilde{\beta}^t).$$

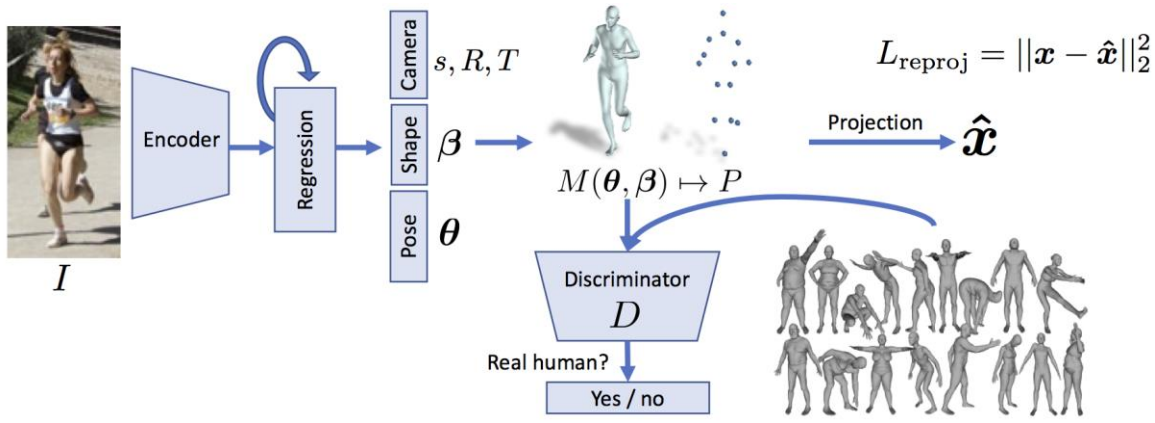
[1] Bogo *et al.* Keep it SMPL ... *ECCV* 2016.

Bundle Adjustment

- Objective function consists of reprojection, temporal and prior terms

$$E(\beta, \theta, \mathbf{\Omega}) = E_R(\beta, \theta, \mathbf{\Omega}) + E_T(\beta, \theta, \mathbf{\Omega}) + E_P(\theta, \beta)$$

- Use the per-frame results of HMR [1] to initialise.
- Optimise with L-BFGS
- Only optimising SMPL- and camera-parameters
- $10 + 75F$ parameters where F is the number of frames in the video



[1] Kanazawa *et al*. End-to-end recovery of human shape and pose. CVPR 2018.

Scaling up to Kinetics

- Data very noisy.
- Initialisation from HMR and 2D pose detector often incorrect.
- Also need to deal with multiple people
- Tracking person-of-interest not robust to detection failures.
- Modify reprojection loss instead

$$E_R(\beta, \theta^t, \mathbf{\Omega}^t) = \min\left(\min_{p \in P^t} \sum_i^J w_i h(\mathbf{x}_i^t - \mathbf{x}_{det,i}^{t,p}), \tau_R\right)$$

Scaling up to Kinetics

- Data very noisy.
- Initialisation from HMR and 2D pose detector often incorrect.
- Also need to deal with multiple people
- Modify reprojection loss instead

$$E_R(\beta, \theta^t, \mathbf{\Omega}^t) = \min\left(\min_{p \in P^t} \sum_i^J w_i h(\mathbf{x}_i^t - \mathbf{x}_{det,i}^{t,p}), \tau_R\right)$$

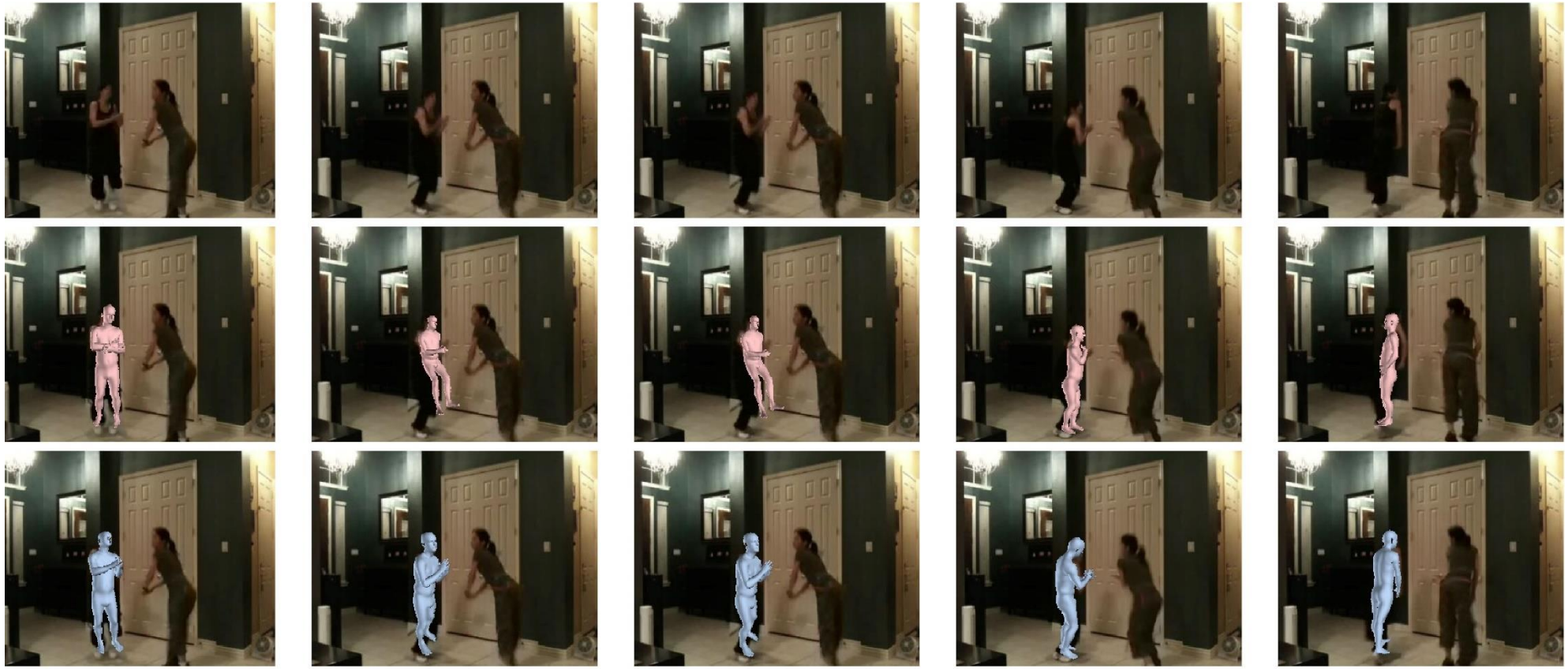
- "Inner min" means the loss is with respect to the best matching 2D pose estimate
- "Outer min" means that if our estimate is too far from 2D pose, we consider it an outlier and pay a constant penalty.

Exploiting temporal consistency

Input

State-of-art HMR model [1] (per-frame)

Bundle adjustment



[1] Kanazawa *et al.* End-to-end recovery of human shape and pose. CVPR 2018.

Exploiting temporal consistency

Input

HMR mesh overlay

Bundle adjustment mesh overlay

Ablation study on Human 3.6M

- Mocap dataset, has metric 3D ground truth
- Allows us to set hyperparameters.
- Each term in objective improves result.
- Ground truth keypoints provide substantial benefits
  - Occluded keypoints help a lot

Method	MPJPE (mm)	PA-MPJPE (mm)
HMR initialisation [20]	85.8	57.5
E_R	154.3	99.7
$E_R + E_P$	79.6	55.3
$E_R + E_P + E_T$	77.8	54.3
E_R (gt. keypoints)	89.2	64.5
$E_R + E_P$ (gt. keypoints)	66.5	45.7
$E_R + E_P + E_T$ (gt. keypoints)	63.3	41.6

Comparison on Human 3.6M

- Compare to other methods using the SMPL model
- Our whole-video approach also achieves state-of-the-art performance on Human 3.6M

Method	MPJPE (mm)	PA-MPJPE (mm)
Self-Sup [49]	–	98.4
Lassner *et al*. direct fitting [23]	–	93.9
SMPLify [7]	–	82.3
Lassner *et al*. optimisation [23]	–	80.7
Pavlakos *et al*. [36]	–	75.9
NBF [32]	–	59.9
MuVS (Note uses 4 cameras) [16]	–	58.4
HMR [20]	88.0	56.8
Ours	**77.8**	**54.3**

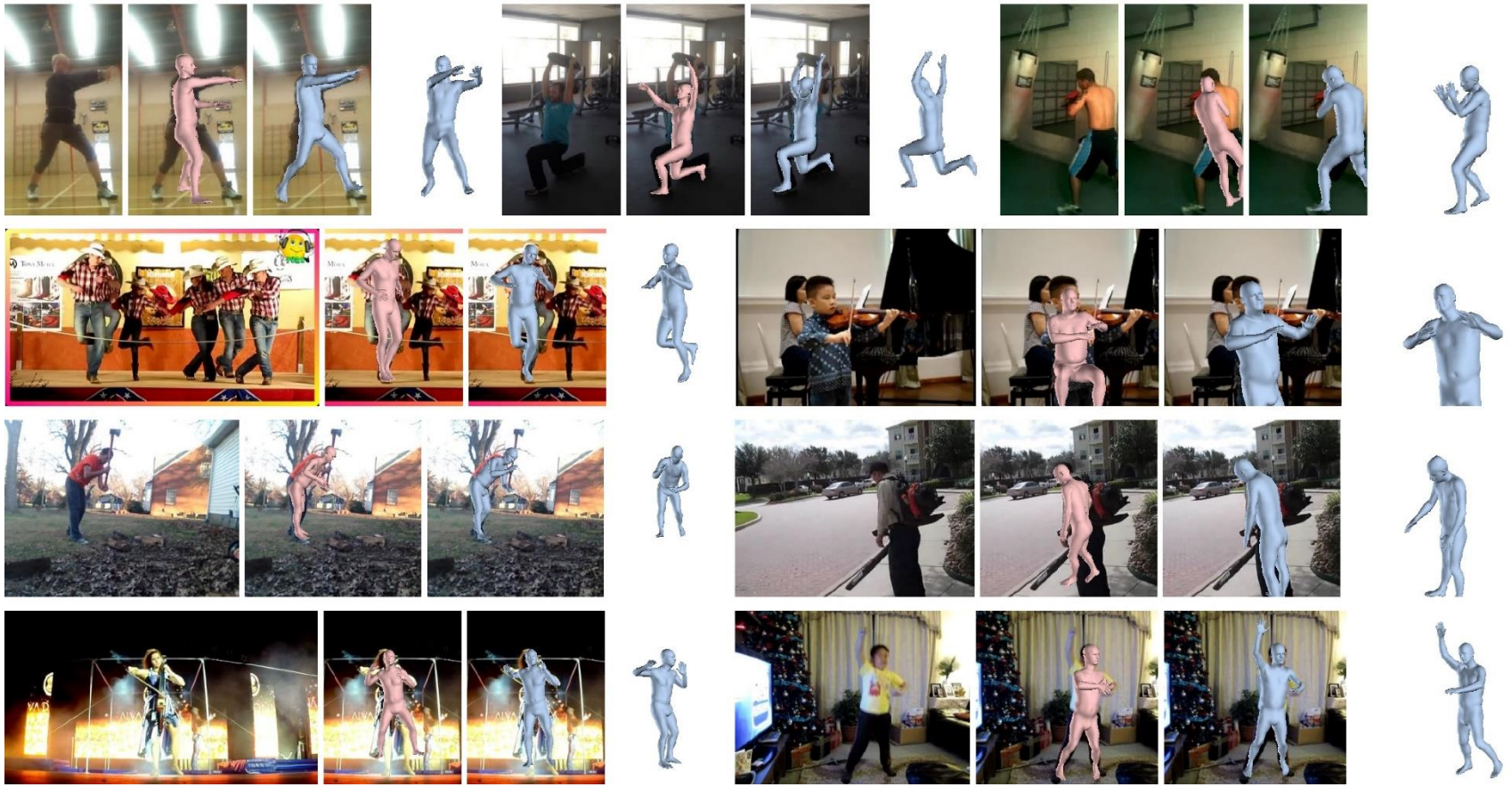
Scaling up to Kinetics

- Run our method on 106 589 YouTube videos in the Kinetics dataset.
- Thresholding the normalised loss, we obtain 16 720 videos containing 4.1 million frames.
- We keep the frames where our 2D projections match that of our 2D keypoint detector.
- Final dataset is 3.4 million frames.
- Available to public: https://github.com/deepmind/Temporal-3D-Pose-Kinetics

Example of video automatically filtered out

Scaling up to Kinetics

- Auto-generated dataset contains diversity in pose, scene, action and camera



HMR (per-frame model)

Ours

Effect of our new dataset

- Training with our new automatically-generated dataset improves the performance of HMR on two datasets.
- 3DPW – “in-the-wild”
- HumanEVA – mocap
- More improvement from 3.4 million additional frames, than 300 000 frames.

PA-MPJPE error (mm) on 3DPW and HumanEVA datasets

Dataset	Original data	Original + Kinetics 300K	Original + Kinetics 3M
3DPW	77.2	73.8	**72.2**
HumanEVA	85.7	83.5	**82.1**

Meta-Learning Deep Visual Words for Fast Video Object Segmentation

Harkirat Behl, Mohammad Najafi,
Anurag Arnab, Philip Torr

One Shot Video Object Segmentation

- Given: Ground-truth masks of multiple objects in the first frame

t=1

One Shot Video Object Segmentation

- Given: Ground-truth masks of multiple objects in the first frame
- Goal: Segment them in rest of the video

t=1

t=T

One Shot Video Object Segmentation

- Given: Ground-truth masks of multiple objects in the first frame
- Goal: Segment them in rest of the video
- Aim: Adapt system to new objects and scene

t=1

t=T

Fine-tuning based Approaches

- Aim: Adapt system to new objects and scene
- Current approach for adapting = Fine-tuning:

Fine-tuning based Approaches

- Aim: Adapt system to new objects and scene
- Current approach for adapting = Fine-tuning:
  - Take a pre-trained segmentation network and fine-tune on ground-truth masks of objects in first frame [1]
  - Some methods perform further online fine-tuning to adapt better to the objects of interest [2]

[1] Caelles et al. *CVPR* 2018 . [2] Li et al. ECCV 2018

Fine-tuning based Approaches

- Drawbacks of Fine-tuning:
  - Very time consuming (~700s to 3h per video)
  - Best performing methods on DAVIS challenge take about ~15s/frame [3]
- Our Aim: Fast adaptation to new objects and scene
  - No finetuning!
  - A single forward pass to segment multiple objects

[3] Bao et al. *CVPR* 2018.

Metric-learning based Approaches

- Our Aim: Fast adaptation to new objects and scene.
- Metric Learning:
  - Embed pixels from same object close to each other in a learned embedding space
  - Embed pixels from different objects far apart

Metric-learning based Approaches

Nearest Neighbour based

- Represents each class with an index of embeddings of all pixels [4]
- Pixels in subsequent frames classified with nearest neighbours
- Greater modelling capacity
- Computationally very expensive ; poor scaling

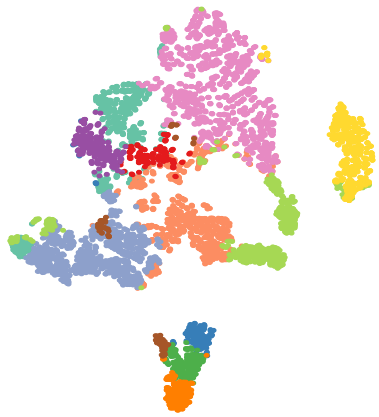
Prototype based

- Represent each class with the mean of their embeddings [5]
- Pixels in subsequent frames classified with softmax over distances to each prototype
- Insufficient capacity to model complex, multi-modal distributions
- Fast

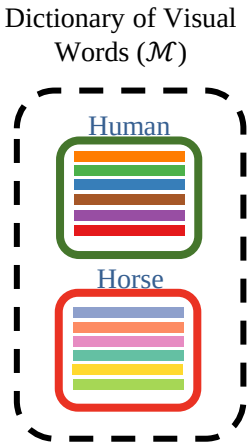
[4] Chen et al. *CVPR* 2018. [5] Snell et al. *NIPS* 2017.

Our Approach

- Represent object as a set of cluster centroids in a learned embedding space
- Fast: Only have to match to cluster centroids
- Accurate: Different cluster centres can model occlusions, deformations, reappearances of objects
- Cluster centres, or “visual words” in embedding space correspond to object parts in image space.



t-SNE visualization of embedding space

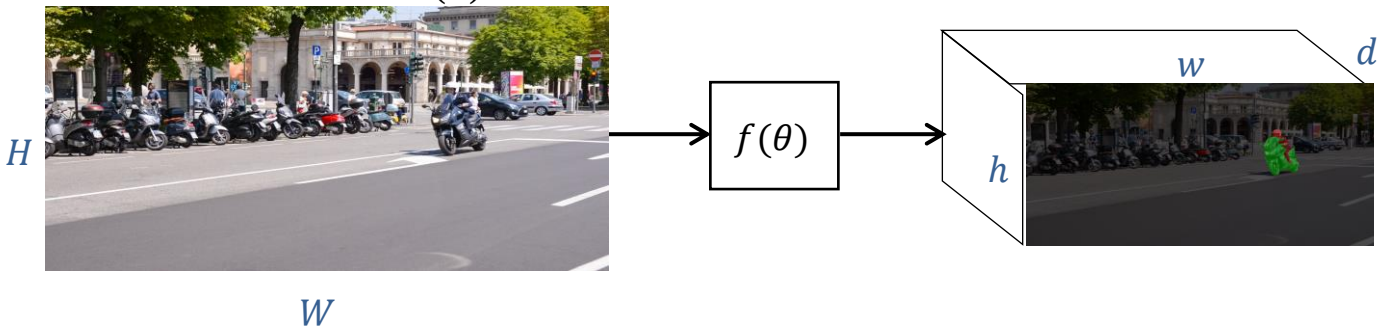


Efficient Object Representation

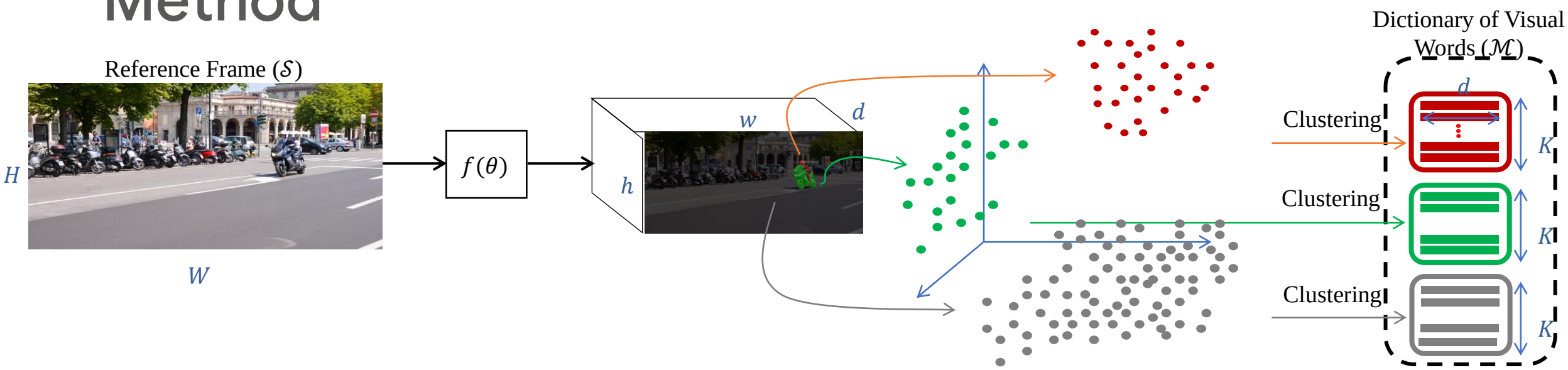
- Represent object as a set of cluster centroids in a learned embedding space
- Cluster centres, or “visual words” in embedding space correspond to object parts in image space.
- Not straightforward, because:
  - No ground truth labels for object parts
  - No ground truth labels for assignment of pixels to object parts.

Method

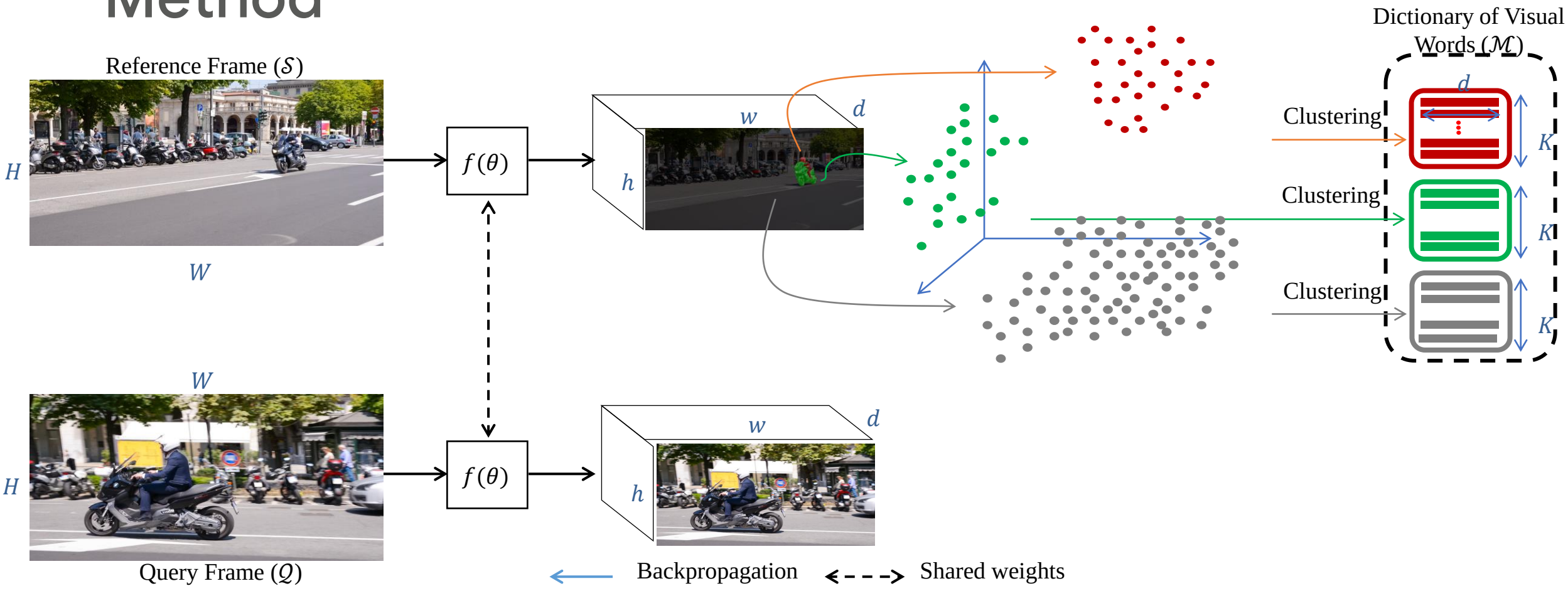
Reference Frame ($\mathcal{S}$)



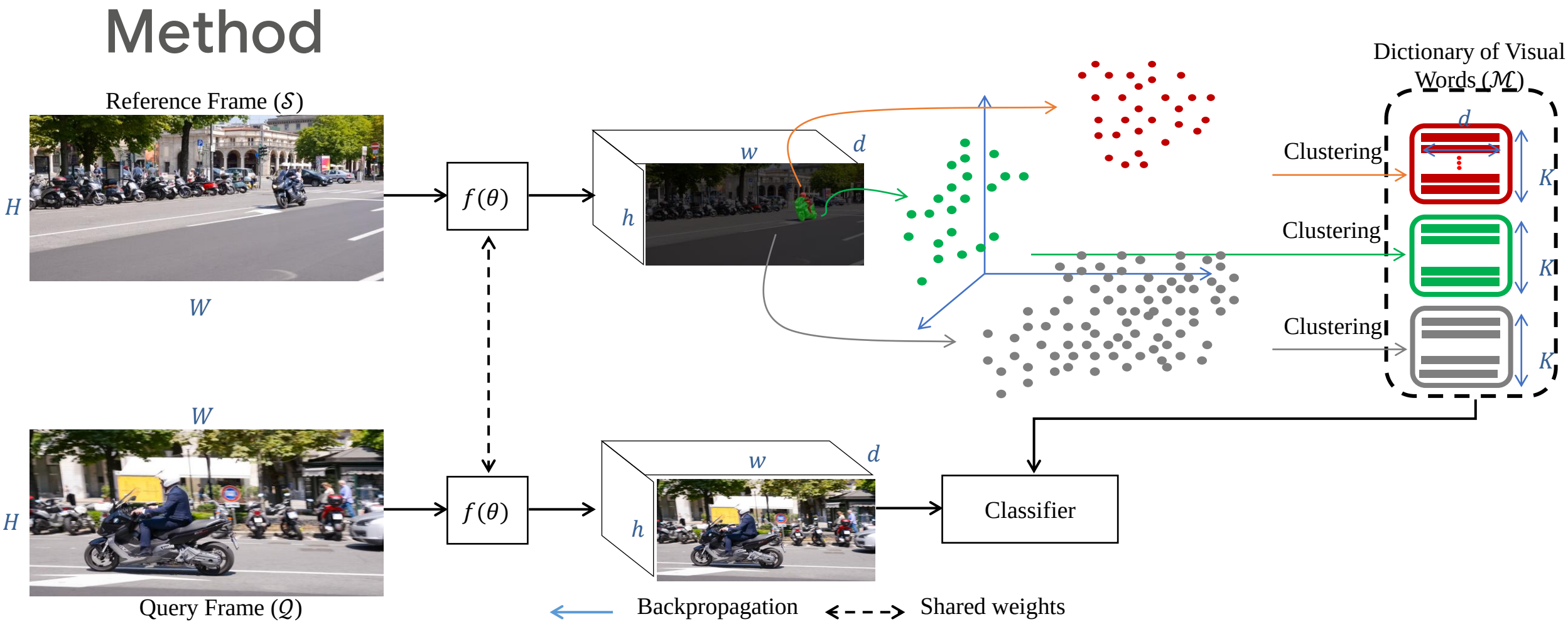
Method


Reference Frame ($\mathcal{S}$)
H
W
$f(\theta)$
h
w
d
Dictionary of Visual Words ($\mathcal{M}$)
Clustering
Clustering
Clustering
d
K
K
K

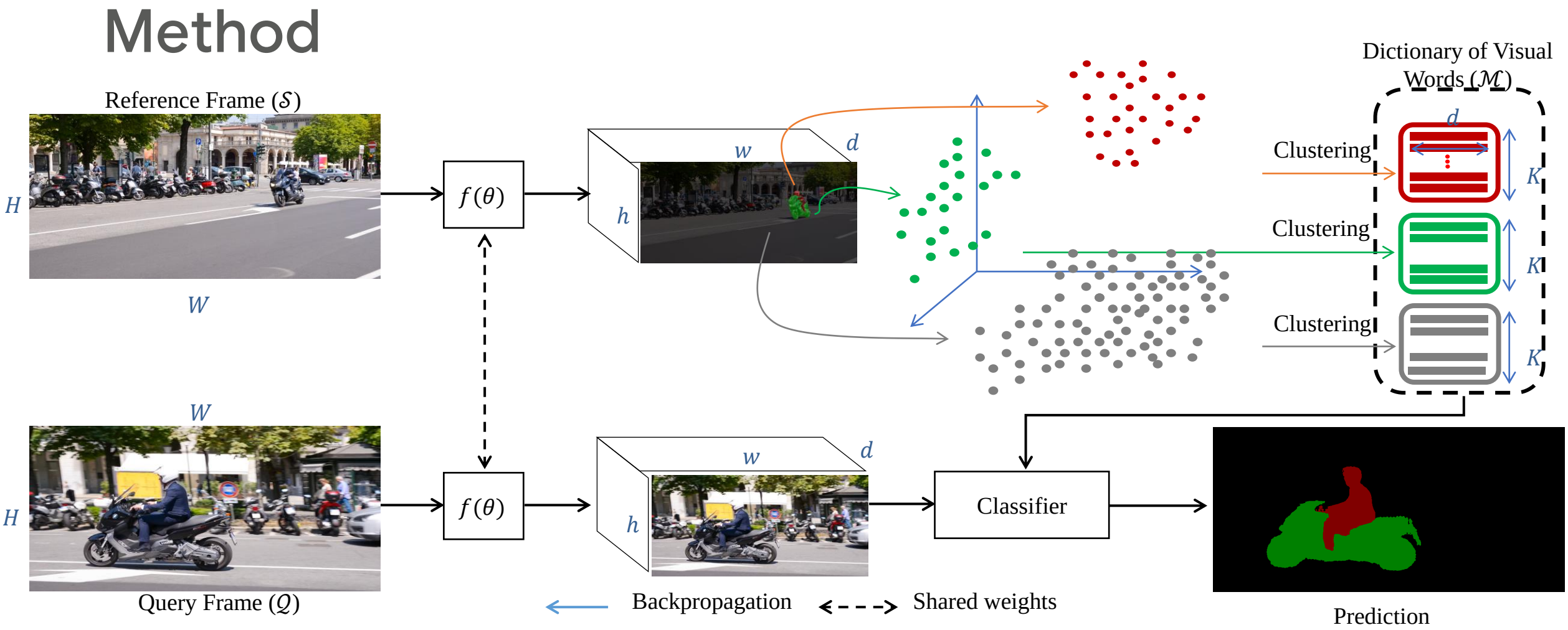
Method


Reference Frame ($\mathcal{S}$)
H
W
$f(\theta)$
h
w
d
Dictionary of Visual Words ($\mathcal{M}$)
Clustering
Clustering
Clustering
K
K
K
W
H
Query Frame ($\mathcal{Q}$)
$f(\theta)$
h
w
d
Backpropagation
Shared weights

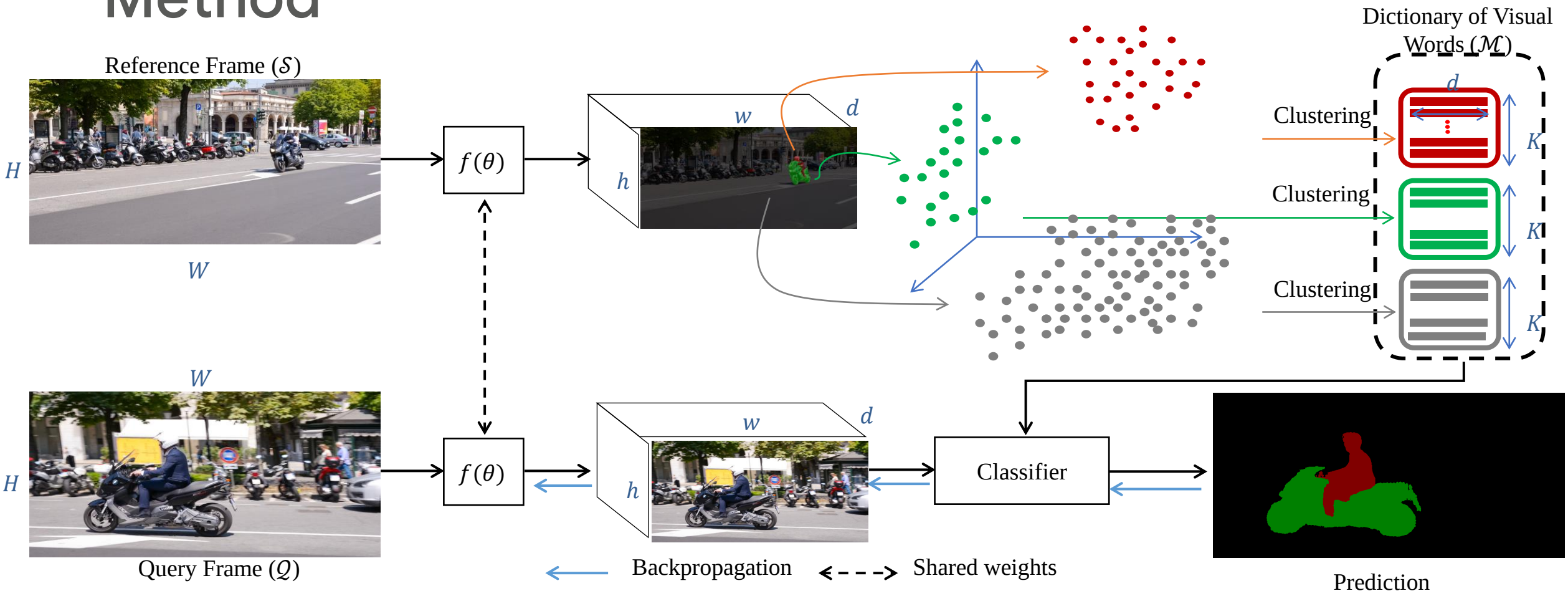
Method


Reference Frame ($\mathcal{S}$)
H
W
$f(\theta)$
h
w
d
Dictionary of Visual Words ($\mathcal{M}$)
Clustering
Clustering
Clustering
K
K
K
W
H
$f(\theta)$
h
w
d
Classifier
Query Frame ($\mathcal{Q}$)
Backpropagation
Shared weights

Method


Reference Frame ($\mathcal{S}$)
H
W
$f(\theta)$
h
w
d
Dictionary of Visual Words ($\mathcal{M}$)
Clustering
Clustering
Clustering
K
K
K
W
H
$f(\theta)$
h
w
d
Classifier
Query Frame ($\mathcal{Q}$)
Backpropagation
Shared weights
Prediction

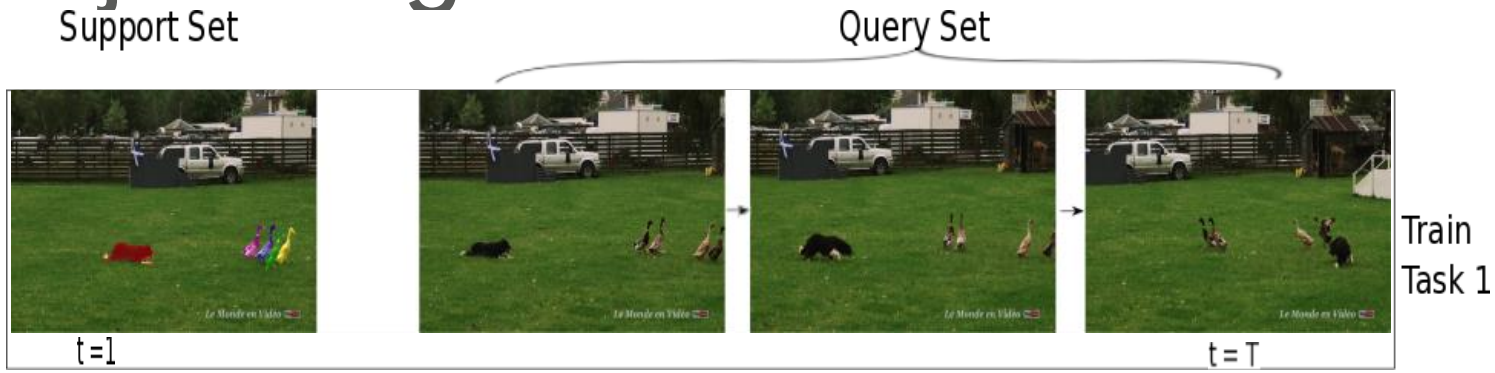
Method


Reference Frame ($\mathcal{S}$)
H
W
$f(\theta)$
h
w
d
Dictionary of Visual Words ($\mathcal{M}$)
Clustering
Clustering
Clustering
K
K
K
W
H
Query Frame ($\mathcal{Q}$)
$f(\theta)$
h
w
d
Classifier
Backpropagation
Shared weights
Prediction

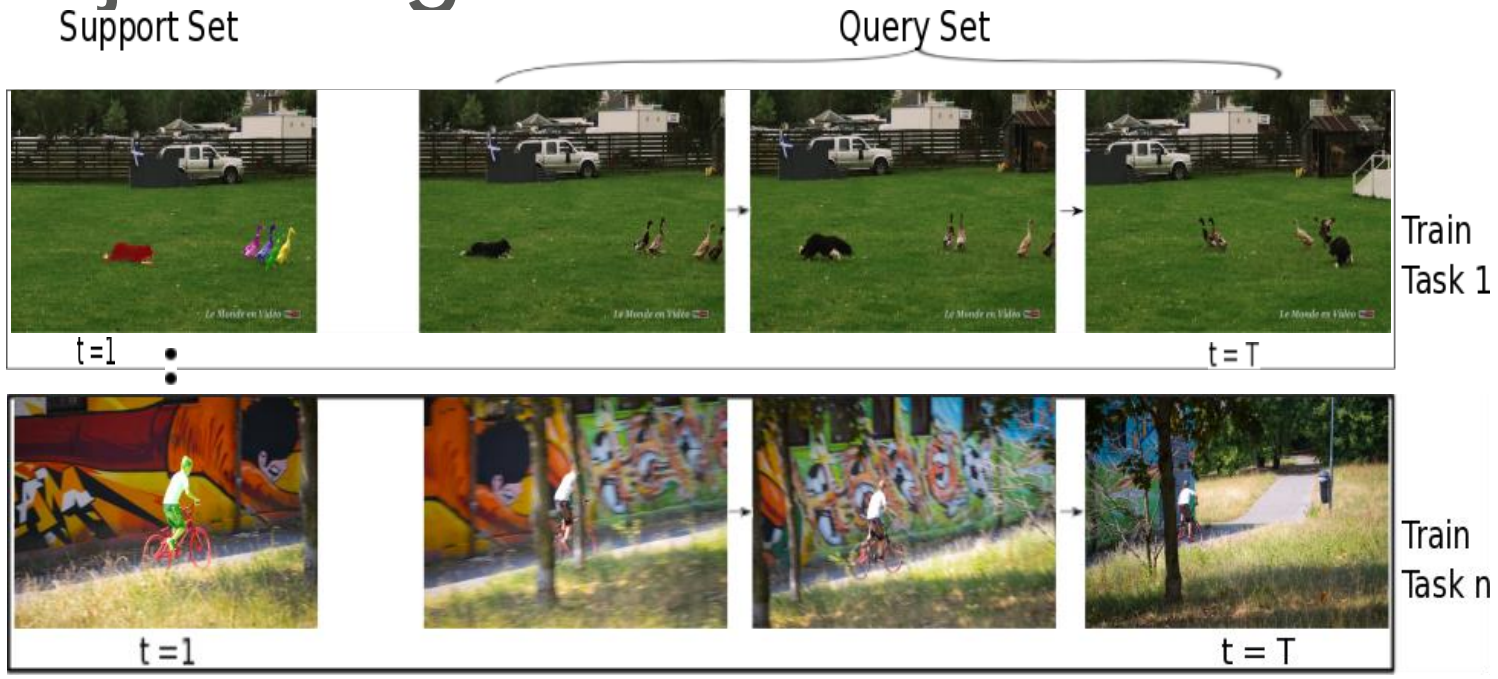
Video Object Segmentation as Meta Learning

- Meta learning: learning from a number of tasks in the training set, to become better at learning a new task in the test set
- Single task:
  - Learn from ground-truth masks of objects in first frame (support set)
  - To segment and track them in rest of video (query set)
  - Each video presents a new task

Video Object Segmentation as Meta Learning



Video Object Segmentation as Meta Learning



Video Object Segmentation as Meta Learning

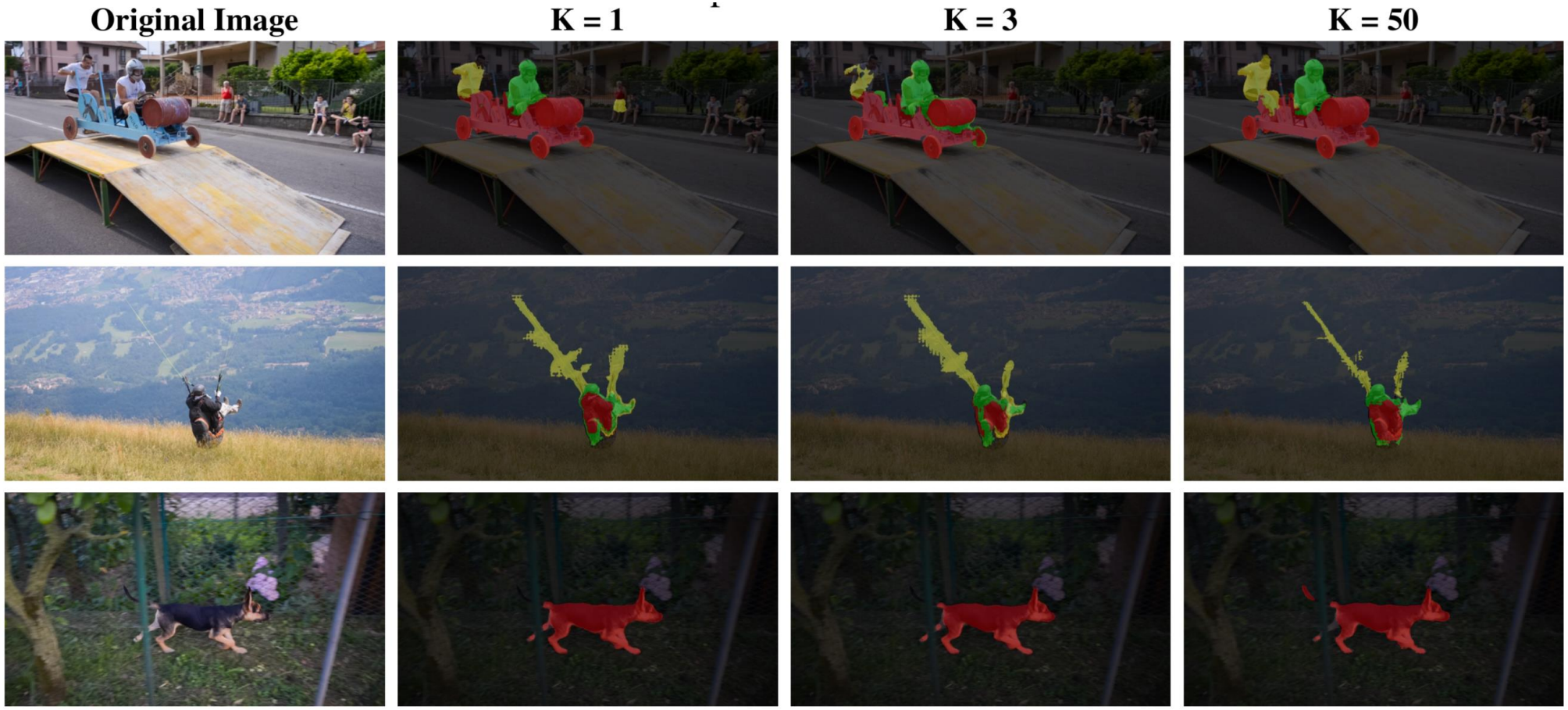
Video Object Segmentation as Meta Learning

Online Update

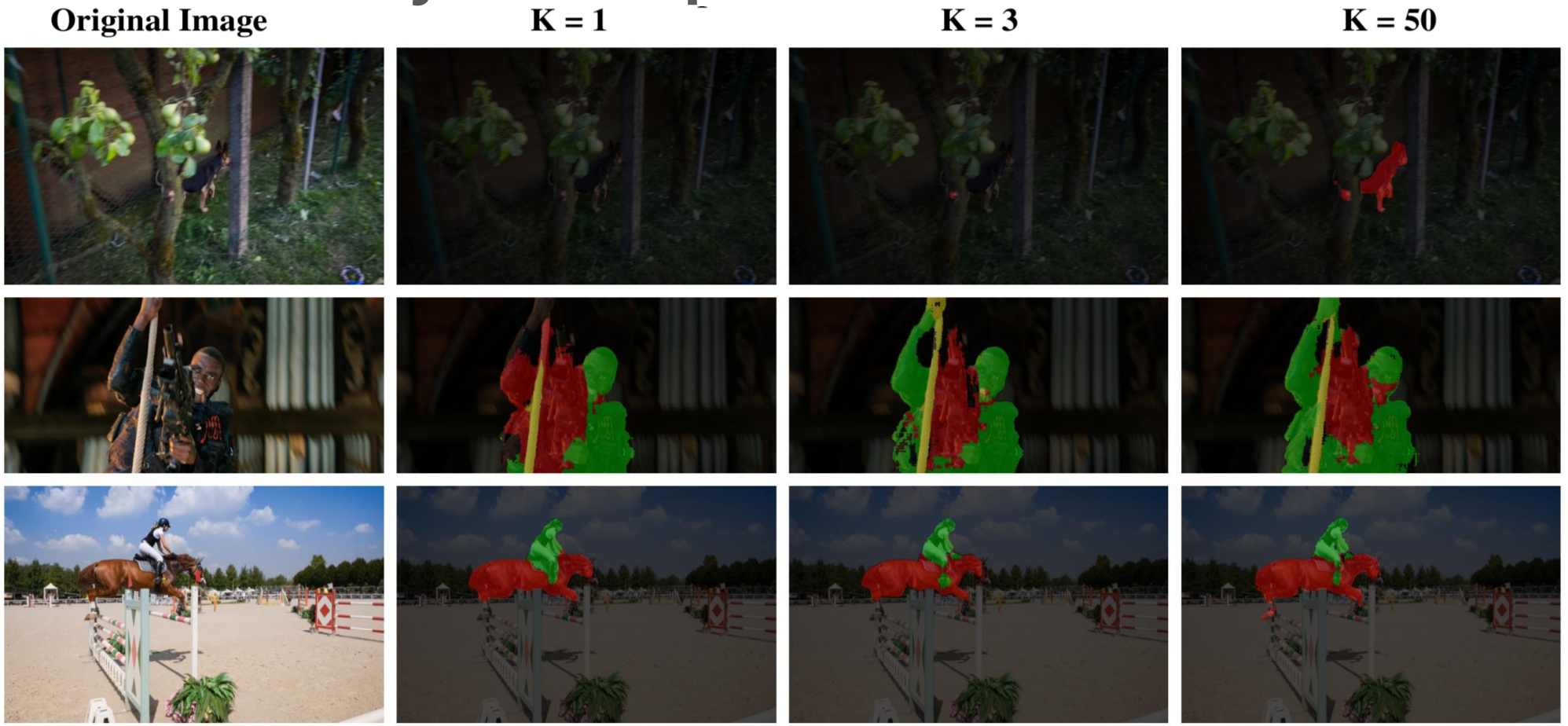
- As the object pose changes over time:
  - Update dictionary of visual words that represent objects
  - Matching will be done to the new visual words/parts
  - None of the existing visual words are discarded

Online Update

Efficient Object Representation



Efficient Object Representation

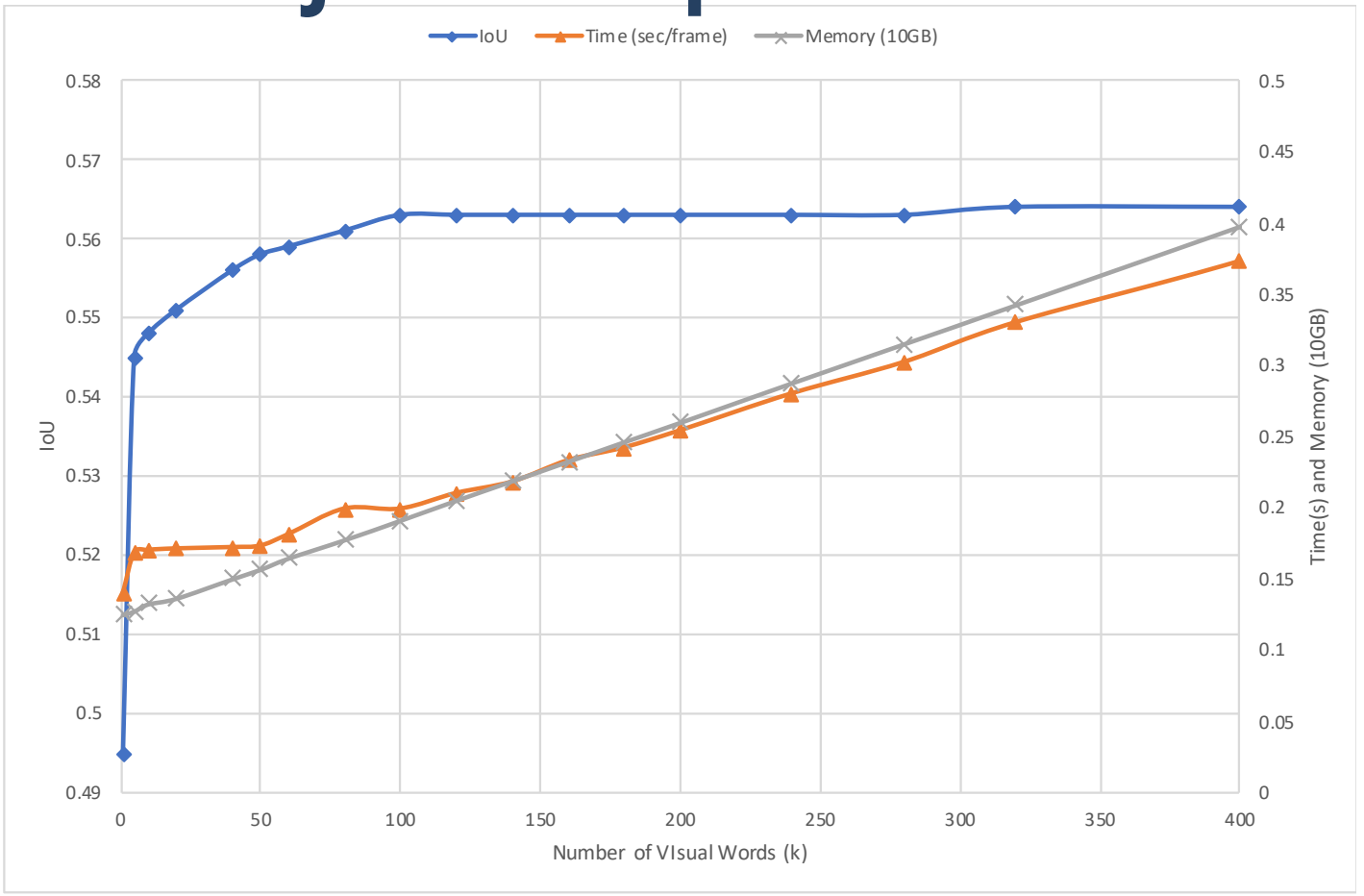


Efficient Object Representation

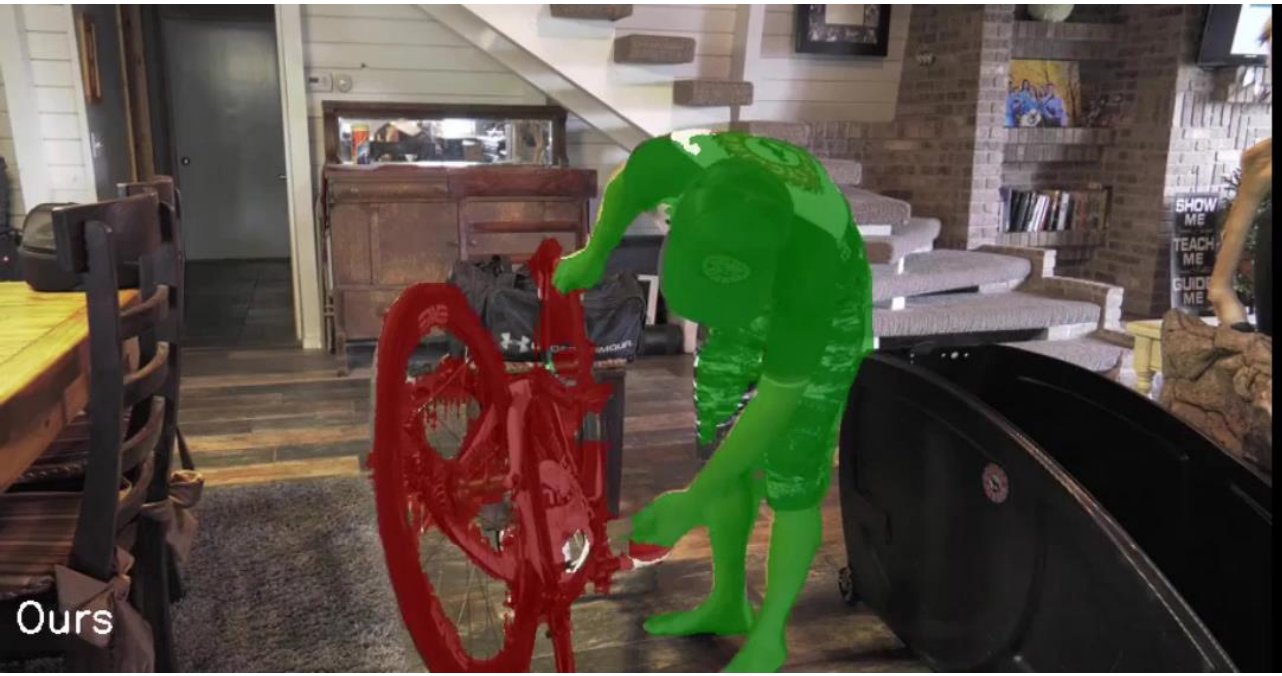
Table 2: **The effect of different object representations** The same MS-COCO pretrained network is used, without any online adaptation. We use the 5 nearest neighbours, following [6].

Model	$\mathcal{J}(\%)$	Time(s)
Single prototype	32.9	**0.14**
5 Nearest neighbours	45.9	5.50
Deep visual words ($k = 50$)	**48.4**	0.17

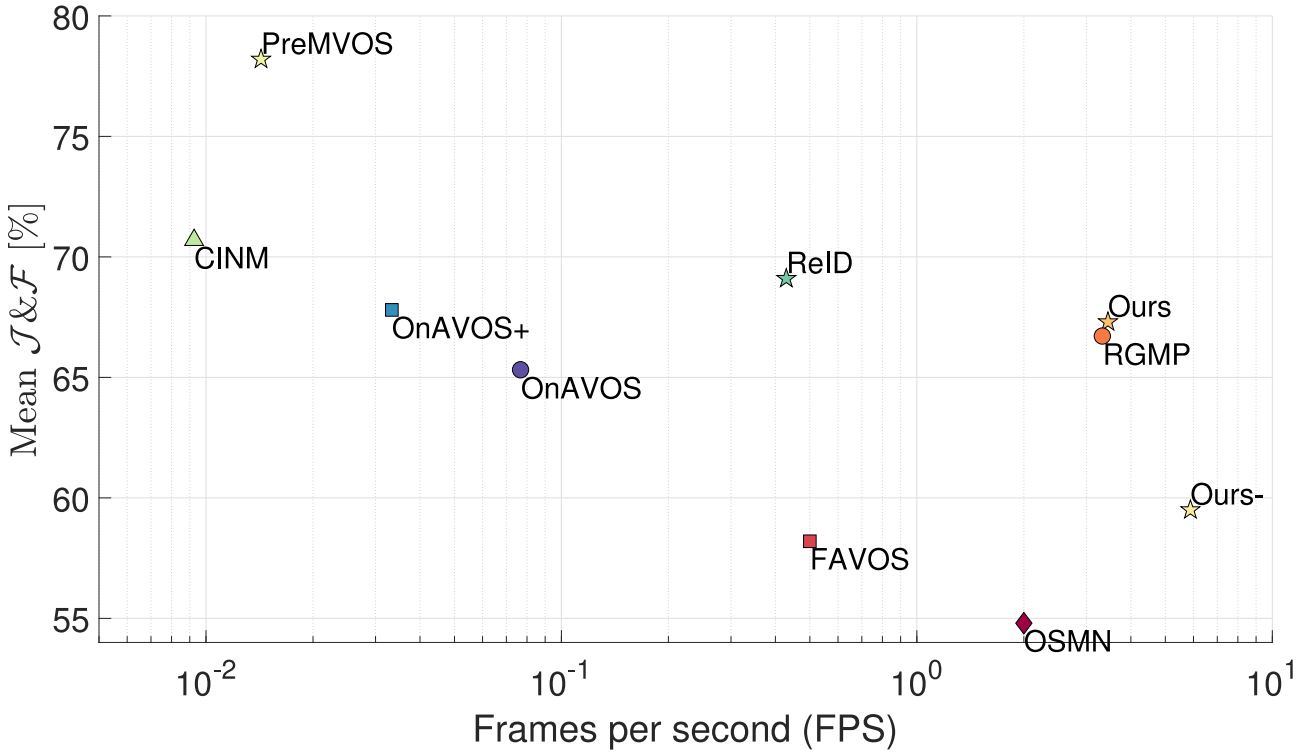
Efficient Object Representation



Results



Accuracy vs Speed



Results - DAVIS-17

Method	FT	PP	OF	$\mathcal{J}(\%)$	$\mathcal{F}(\%)$	$\mathcal{J}\&\mathcal{F}(\%)$	Time(s)
MaskRNN [19]	✓		✓	60.5	–	–	9s
OSMN [49]	✓	✓		60.8	–	–	–
OnAVOS [46]	✓	✓		61.6	69.1	65.3	13s
OnAVOS† [46]	✓	✓		64.5	71.2	67.8	30s
VideoMatch [21]	✓			61.4	–	–	2.62s
ReID [30]	✓		✓	67.3	71.0	69.1	2.33s
OSVOS^S [4]	✓	✓		64.7	71.3	68.0	–
CINM† [2]	✓	✓	✓	67.2	74.4	70.7	$\sim$ 108s
PReMVOS† [23]	✓		✓	**74.3**	**82.2**	**78.2**	$\sim$ 70s
OnAVOS [46]				39.5	–	–	3.78s
MaskRNN [19]			✓	45.5	–	–	0.6s
VideoMatch [21]				56.5	–	–	0.35s
RGMP [47]				**64.8**	68.6	66.7	–
Ours$^-$				55.8	63.1	59.5	0.17s
Ours				63.9	**70.7**	**67.3**	0.29s

Table 1: **Results on DAVIS-2017 validation Dataset.** FT: Fine-Tuning on the first frame of the test video; PP: Post-Processing; OF: Optical Flow; $\mathcal{J}\&\mathcal{F}$: The mean of $\mathcal{J}$ and $\mathcal{F}$ metrics; Time(s): The time (in seconds) spent on each frame on average; Ours$^-$: Our model without online adaptation; †: Ensemble of models are used.

Results - YouTube-VOS

Method	FT	PP	OF	$\mathcal{J}(\%)$
MSK [10]	✓	✓	✓	72.6
Lucid [6]	✓	✓	✓	76.2
OnAVOS [12]	✓	✓		77.4
DRL [3]	✓			78.1
OSVOS [2]	✓	✓		78.3
CINM [1]	✓	✓	✓	78.4
ReID [7]	✓		✓	79.6
OSVOS^S [9]	✓	✓		**83.2**
BVS [8]				68.0
OSMN [13]				69.0
VideoMatch [4]				79.7
Ours				**81.1**

Conclusion

- Alternative of fine-tuning for efficient adaptation
- Efficient Object Representation
- Intuitive approach

Uncertainty-Aware Weakly Supervised Action Detection from Untrimmed Videos

Anurag Arnab, Chen Sun,
Arsha Nagrani, Cordelia Schmid

Google Research

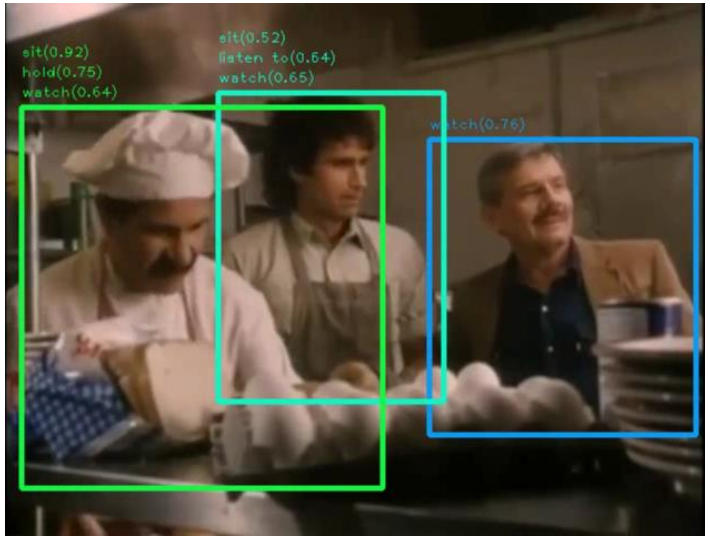
Introduction

- Spatio-temporal action detection
  - Bounding box in space and time around action of interest
- Most approaches extend detectors, such as Faster-RCNN and SSD, temporally.
- In this paper, we only use cheap, video-level labels

Annotation

Labels: *sit, hold, watch, talk to, listen to*

Prediction



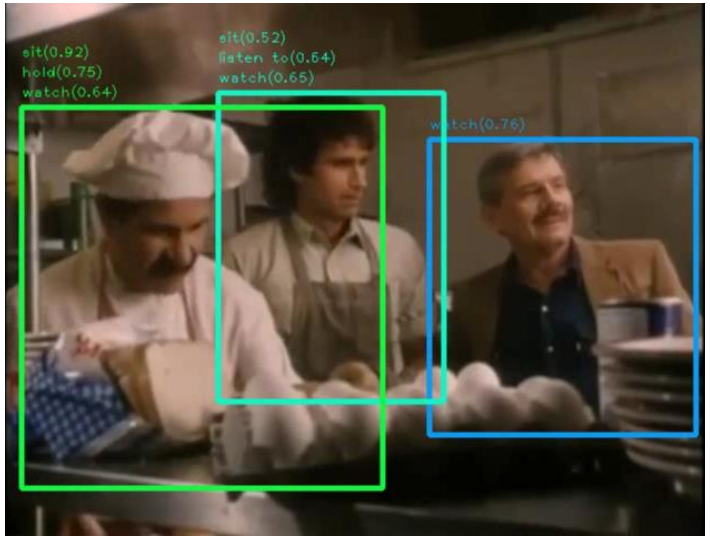
Weaker supervision

- Labelling bounding boxes per frame is too expensive
- Temporal boundaries of actions are ambiguous, annotators often do not agree with each other
- Only use cheap, video-level labels.

Annotation

Labels: *sit, hold, watch, talk to, listen to*

Prediction



Approach Overview

- Leverage off-the-shelf, per-frame person detectors to obtain person tubelets.
- Multiple Instance Learning
  - Each bag is formed from all tubelets in the video
- Due to noise, and violations of the MIL assumptions, predict the uncertainty for each bag as well.

Multiple Instance Learning

- Have a bag of examples, $\mathbf{x} = \{x_1, x_2, \ldots, x_n\}$.
- Only know label for the whole bag, y.
- Key assumption is that one or more instances in the bag have label y.
- Want to train an instance-level classifier.
    - Classify each instance in the bag.

Multiple Instance Learning

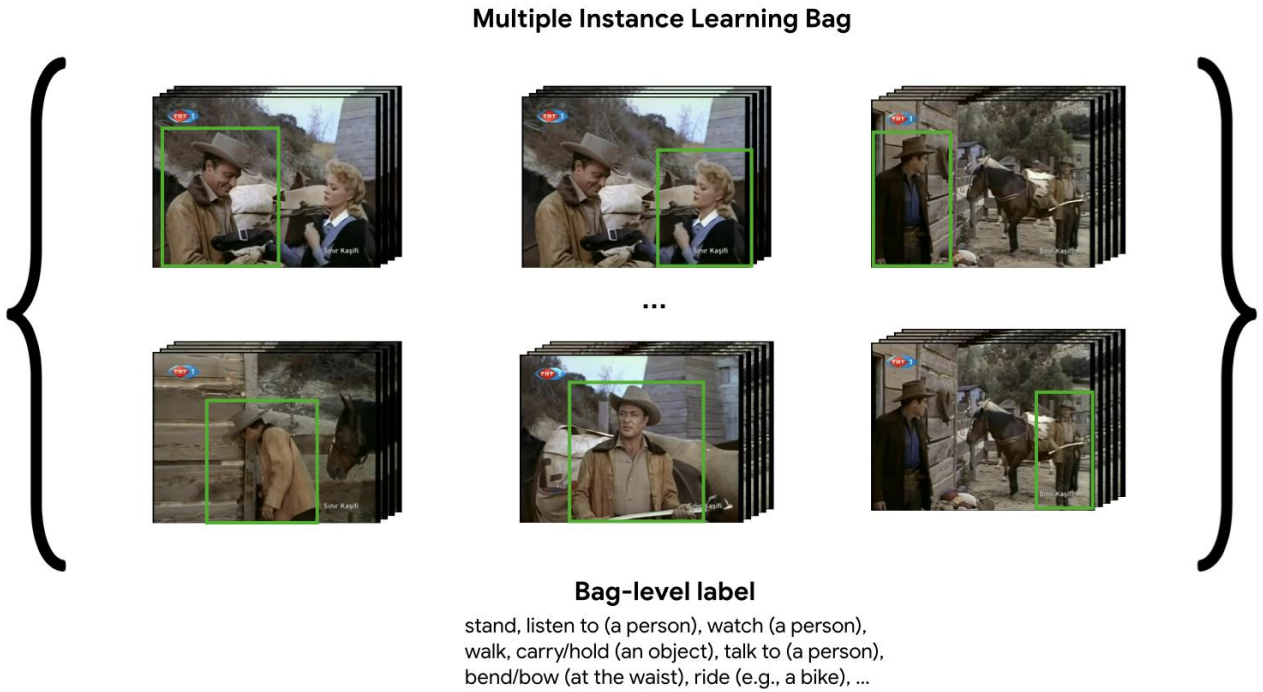
- Have a bag of examples, $\mathbf{x} = \{x_1, x_2, \ldots, x_n\}$.
- Only know label for the whole bag, y.
- Want to train an instance-level classifier.
- Aggregate instance-level predictions into a bag-level prediction.

$$p(y_l = 1 | x_1, x_2, \ldots, x_n) = g(p_i, p_2, \ldots, p_n)$$

- Use standard loss function
- Common aggregation functions: max, log-sum-exp, average, attention

Multiple Instance Learning (MIL)

- All the person tubelets within a video form a “bag”
  - Person tubelets are detections linked over at most K frames.
- The standard MIL assumption is that at least one tubelet in the bag has the video-level label.



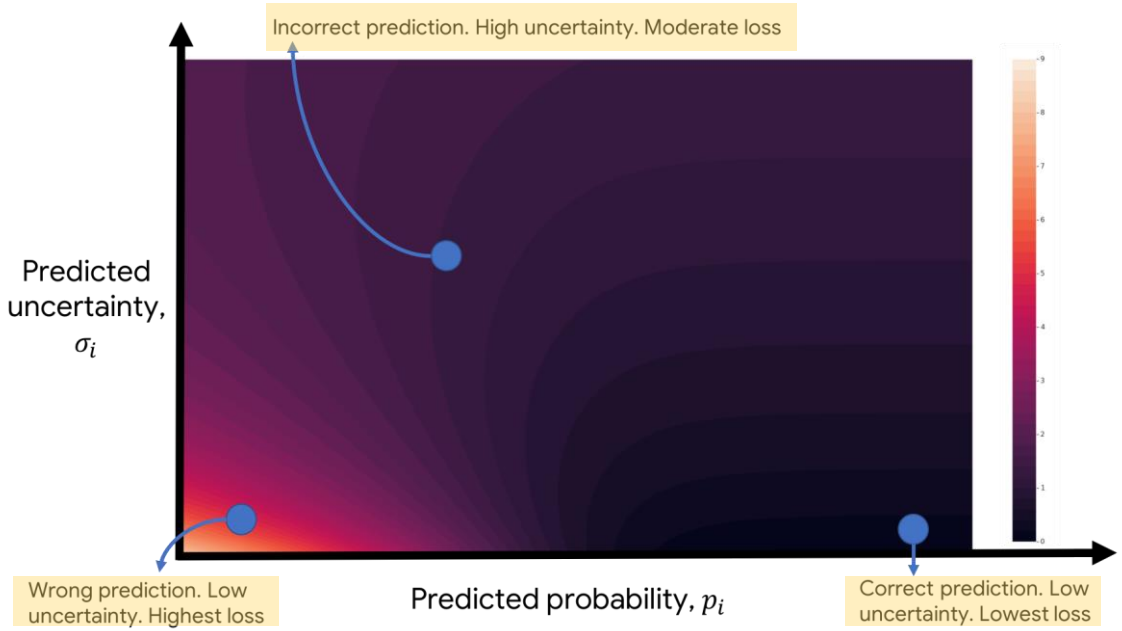
Label Noise and Violations of MIL Assumption

- MIL assumption is often violated
- Sampling bags
  - Cannot fit a whole bag in memory
  - Particularly as videos get longer
  - Uniformly sample tubes
- Person detector errors
  - Due to domain gap
  - False positives as some datasets don’t label actors exhaustively

All person detections besides the pole-vaulter are considered false-positives in this dataset.

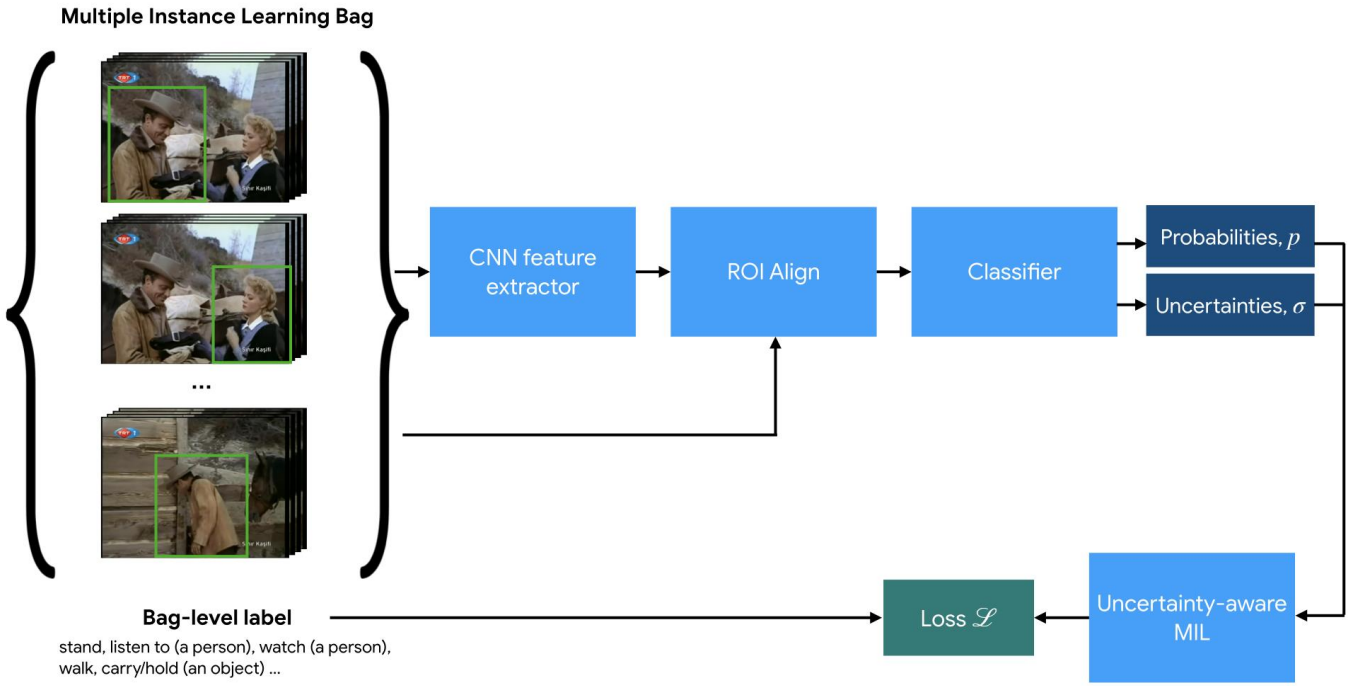
Uncertainty Estimation

- Predict uncertainty for each instance in the bag
- Intuition:
  - When possible, predict correct label with low uncertainty
  - Otherwise, predict incorrect label with high uncertainty.
- $L(x, y, \sigma) = \frac{1}{\sigma^2}\mathcal{L}_{ce}(x, y) + \lambda \log(\sigma^2)$



Network Architecture

- Fast-RCNN style detector
- Use person tubelets as proposals

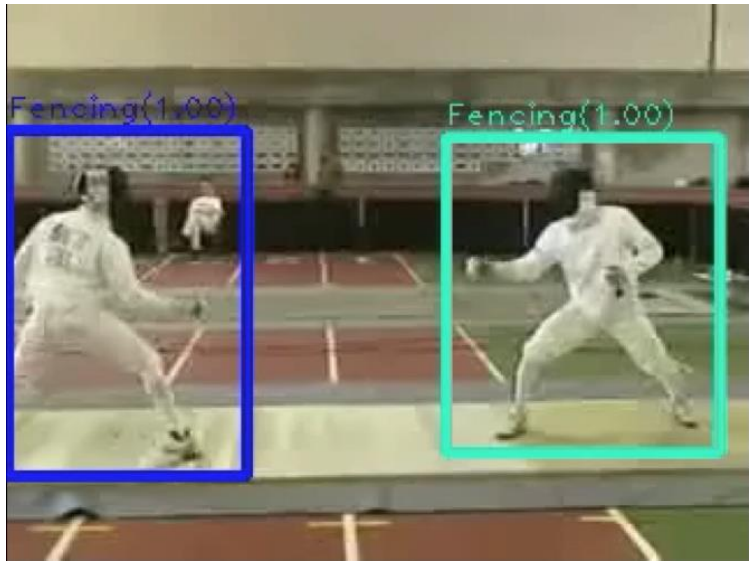


Evaluation Datasets

- UCF101-24
  - Most common dataset
  - Sports videos from YouTube, 24 classes
  - Many “background people” not doing the labelled action
  - Evaluate Video AP
- AVA
  - 60 atomic actions, from 15 minute movie clips
  - Keyframes at 1Hz, are labelled. Predict actions at keyframe given temporal context
  - Evaluate Frame AP

UCF101-24 Ablation

	Video AP	
	0.2	0.5
Weakly supervised baseline	54.3	29.7
MIL - LSE pooling	60.1	33.1
MIL - mean pooling	60.3	33.0
MIL - max pooling	60.7	33.5
MIL - max pooling, uncertainty	61.7	35.0
Fully supervised	69.3	43.6



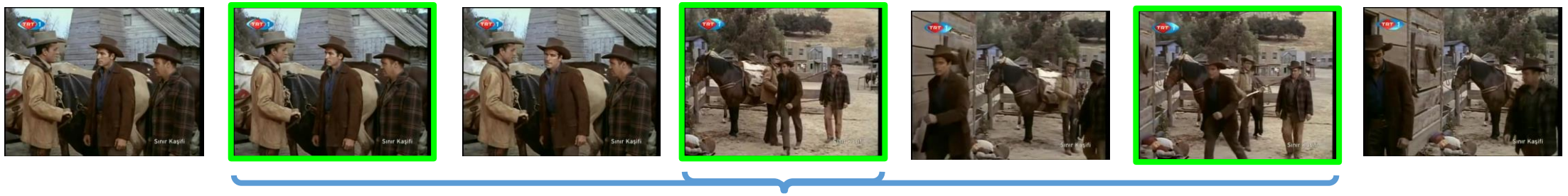
- Big domain gap between COCO and UCF
- Detector, trained only on COCO, has 47% recall and 21% precision on UCF training set.
- Sampling tubelets is necessary: Average of 33.1 tubelets per video, V100 GPU can hold 16.

UCF101-24 Comparison

	Video AP at 0.2	Video AP at 0.5
Fully supervised		
Peng *et al.* [35]	42.3	35.9
Hou *et al.* [17]	47.1	–
Weinzaepfel *et al.* [50]	58.9	–
Saha *et al.* [38]	63.1	33.1
Singh *et al.* [41]	73.5	46.3
Zhao *et al.* [52]	78.5	50.3
Singh *et al.* [40]	79.0	50.9
Kalogeiton *et al.* [19]	77.2	51.4
Ours	69.3	43.6
Weakly supervised		
Escorcia *et al.* [8]	45.5	–
Chéron *et al.* [6]	43.9	17.7
Ours	61.7	35.0

AVA

- AVA labels keyframes at 1Hz, videos are 15 minutes long.
- Vary the subclip of the video from which we take clip-level annotation
- Problem gets harder as the subclip duration is increased.



AVA Results

	Sub-clip duration (seconds)						
	FS	1	5	10	30	60	900
Frame AP	24.9	22.4	18.0	15.8	11.4	9.1	4.2

Conclusion

- Weakly-supervised spatio-temporal action detection with Multiple Instance Learning
- Predict uncertainty to better handle noise and violations of standard MIL assumption.

Collaborators

- Carl Doersch
- Andrew Zisserman

- Harkirat Behl
- Philip Torr

- Chen Sun
- Arsha Nagrani
- Cordelia Schmid

Questions?

- anurag.arnab@gmail.com

- A Arnab*, C Doersch*, A Zisserman. *Exploiting Temporal Context for 3D Human Pose Estimation in the Wild*. CVPR 2019.
- H Behl, M Najafi, A Arnab, P Torr. *Meta-Learning Deep Visual Words for Video Object Segmentation*. IROS 2020
- A Arnab, C Sun, A Nagrani. C Schmid. *Uncertainty-Aware Weakly Supervised Action Detection from Long Videos*. ECCV 2020