# IPST: Incremental Pictorial Structures for Model-free Tracking of Deformable Objects

Grigorios G. Chrysos, Epameinondas Antonakos, and Stefanos Zafeiriou, *Member, IEEE*

***Abstract*—Model-free tracking is a well-studied task in computer vision. Typically, a rectangular bounding box containing a single object is provided in the first (few) frame(s) and then the method tracks the object in the rest frames. However, for deformable objects (e.g. faces, bodies) the single bounding box scenario is suboptimal; a part-based approach would be more effective. The current state-of-the-art part-based approach is incrementally trained discriminative Deformable Part Models (DPM). Nevertheless, training discriminative DPMs with one or a few examples poses a huge challenge. We argue that a generative model is a better fit for the task. To that end, we utilise the powerful pictorial structures, which we augment with incremental updates to account for object adaptations. Our proposed incremental pictorial structures, which we call IPST, are experimentally validated in different scenarios. In a thorough experimentation we demonstrate that IPST outperforms the existing model-free methods in facial landmark tracking, body tracking, animal tracking (newly introduced to verify the strength in ad hoc cases).**

***Index Terms*—Part-based model-free tracking, Pictorial Structures, Incremental tracking**

## I. Introduction

VISUAL object tracking is among the fundamental problems of computer vision, with a plethora of applications including video surveillance, human computer interaction, augmented reality etc. The past few years considerable improvement has been achieved in model-free tracking of single targets however tracking deformable objects in unconstrained conditions remains challenging.

Model-free trackers aim at estimating the position of an object in a video. No prior information is available about the object; the sole input to the method is the bounding box style annotation(s) (that contain the object) provided for the first few frames [1], [2]. Then, the method tracks the object in the remaining frames. Model-free trackers can be divided into a) holistic, b) part-based methods. The holistic ones represent the object with a single bounding box[1], while the part-based methods explicitly include a set of parts that are tracked. Part-based tracking is well suited for deformable objects where semantic parts might move differently.

The category of holistic methods is predominant while the related literature is extensive. The state-of-the-art mainly revolves around methods that (a) use a tracking-by-detection strategy that treats the tracking problem as a classification task using online learning techniques to update the object model [3], [4], (b) learn Correlation Filters over image features [5], [6] and (c) learn an appropriate function, using Deep Convolutional Neural Networks, which matches the initial target to candidates in all the other frames [7], [8]. For a thorough comparison the interested reader may consult [2], [7], [9].

In rare cases part-based and object shape representations are implicitly used in holistic model-free tracking [10]–[12]. The tracker of Adam *et al.* [10] represents the object with multiple arbitrary patches. Each patch votes on potential positions and scales of the object and a robust statistic is employed to minimise the error from voting. Kalal *et al.* [11] perform part-based sampling; each part (point) is tracked independently in each frame by estimating the optic flow. Using a forward-backward measure, the erroneous points are identified and the rest of the reliable points are utilised to compute the optimal object trajectory. Wang *et al.* [12] introduce a tracker predicting the direct object displacement. A cascade of regressors is utilised to localise the parts, while the model is updated online and the regressors are initialised by a multiple motion model at each frame. The motion model initialisation along with the sensitivity of the regressors can cause drifting in case of rapid movement or severe deformations. Although, the aforementioned trackers use the notion of parts/deformation, their goal is not to track the deformations of the object but rather to make the rigid object tracking model stable to occlusions.

The second category, i.e. part-based (deformable) tracking, has significant applications, however it is relatively understudied, because creating databases with ground-truth part annotations is a gargantuan task. For a handful of well-studied objects, e.g. human face, human body, benchmarks exist; for the majority of objects "in-the-wild" there is no work yet. The annotations of existing benchmarks are typically extracted in a semi-automatic way [13], [14] by building statistical models in carefully annotated datasets. Training model-based fitting methods for arbitrary objects is currently infeasible, since annotating arbitrary objects with regards to parts forms an expensive and tedious process. However, to validate our approach in an ad hoc case, we have annotated a new dataset that includes cats' faces.

The most relevant work to ours is the structure-preserving object tracker (SPOT) [4] and the tracker of Yao *et al.* [3]. The tracker in [3] adapts the latent SVM formulation [15] for online tracking, by making a first order Markov assumption

G.G. Chrysos is with the Department of Computing, Imperial College London, London, UK, (e-mail: g.chrysos@imperial.ac.uk).

E. Antonakos is with Amazon, Berlin, Germany. He was with Imperial College London when this work was done. (e-mail: antonak@amazon.com)

S. Zafeiriou is with the Department of Computing, Imperial College London, London, U.K. (e-mail: s.zafeiriou@imperial.ac.uk).

[1]Since the main aim is to track a rigid representation of the object, deformations and/or the parts of the object are rarely considered in holistic methods.

on the position of the parts, i.e. the location of each part is searched in the neighboorhood of the part's position in the previous frame. The SPOT tracker on the other hand is a strongly-supervised discriminatively trained Deformable-Part-based Model (DPM) [15] object detector that is incrementally updated. The difference between SPOT and DPM is that the location of the parts of the object are given by the user in the first frame (or few frames) of the video, while in Yao *et al.* [3] the parts are latent variables learnt during the training process, and that the parameters of the model are learnt in an online fashion, rather than from a large collection of annotated data as in DPM. Nevertheless, as a discriminative method SPOT requires many well-annotated data and could be prone to drifting.

In contrast to the aforementioned discriminative part-based trackers, we support that a generative model is well-suited for the task. Motivated by the recent works on Active Appearance Models (AAMs) where it was shown that feature-based Principal Component Analysis (PCA) is as powerful as discriminative models for part-based object fitting [16], [17][2], we illustrate such a case in Fig. 1. The PCA, learnt per-part from pixel intensities, is able to reconstruct accurately the texture. Incremental PCA-based trackers were for several years among the state-of-the-art in model-free tracking methods [18], [19]; nevertheless they are holistic methods. Expressing both the variations of the texture and the shape at the same time with a limited number of PCA components is challenging. Our methodology instead decouples the variations between texture and shape, i.e. we devise a method that disentangles the texture and the shape and is incrementally updated.

In this paper we revisit the original generative pictorial structures [20]. Pictorial Structures are a statistical model that assumes a tree structure for the representations; each node models the texture of a part, while each edge the spatial relationship between parts. Each part has a different distribution; we can model deformable objects whose parts have different (appearance/shape) variance, e.g. the nose tip versus the eyes of a human face. We augment the original formulation to include a) appropriate features (e.g., HoG [21]), b) incremental updates (which are exact). The features consist our method robust to illumination changes that might occur in sequential frames, while the increment updates adjust the model to the object adaptations. Our incremental pictorial structures tracker, IPST for short, is accurate and can track effectively any object if given the annotations of the first few frames. Our thorough experimentation dictates that IPST is better suited for deformable model-free tracking than its discriminative counterpart [4]. This is because in a model-free tracking setting annotated data are scarce and the training algorithm can be sensitive to erroneous fittings. We also compare with tracking all the parts independently using several well-established holistic trackers (which is a computationally expensive process due to the number of parts).

Our contributions are organised as follows:

[2]It is very difficult to apply AAMs in a model-free framework, since they use a non-linear optimisation for fitting which gets easily stack in wrong solutions without the use of a generic model for the object.

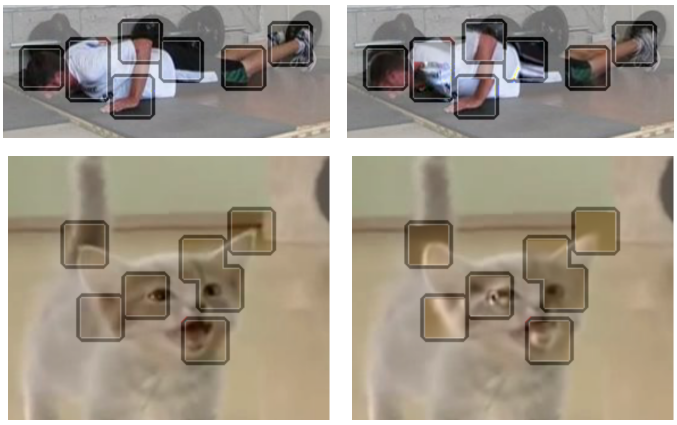


Fig. 1: (best viewed in colour) Visual illustration of the linear yet powerful incremental update of a part-based representation in two videos. **Left:** The $90^{th}$ frame of a video. **Right:** The same frame with the parts (projected and) reconstructed from PCA's. The 20 first frames of each video were used for incrementally updating a per-part PCA. Minor artifacts can be noticed in few parts, e.g. the hand continuity or the reconstruction of the cat's tail/mouth, however each part is still comprehensible in unseen future frames (e.g. 70 frames ahead).

- We introduce IPST, a part-based (deformable) model-free tracker which combines the powerful pictorial structures with incremental updates.
- In a thorough experimentation we demonstrate that IPST outperforms the existing trackers in several benchmarks. In addition, to emphasise the ad-hoc merit of IPST, we annotated a new dataset with animals. This dataset will be released upon the acceptance of the paper.
- We release an open-source implementation of IPST, which can be very useful for initialising ad-hoc deformable tracking cases.

## II. Method

For a sequence of frames, the goal is to track $n$ points in each frame. In model-free tracking, there is no a-priori knowledge of the $n$ points of interest, hence they are provided with manual annotation in the first $\mathcal{K}_0$ images. We model both the appearance and the spatial relationship of the $n$ points with Gaussian distributions (as originally done by Felzenszwalb *et al.*in [20]), while we update both models in an incremental manner to account for object adaptations; a visual illustration of the proposed system exists in Fig. 2. In the subsequent sections we describe the model, the parameter learning and updates along with the derivation of the cost function.

### A. Notation

A capital (small) bold letter denotes a matrix (vector) representation, a plain letter represents a scalar number. The

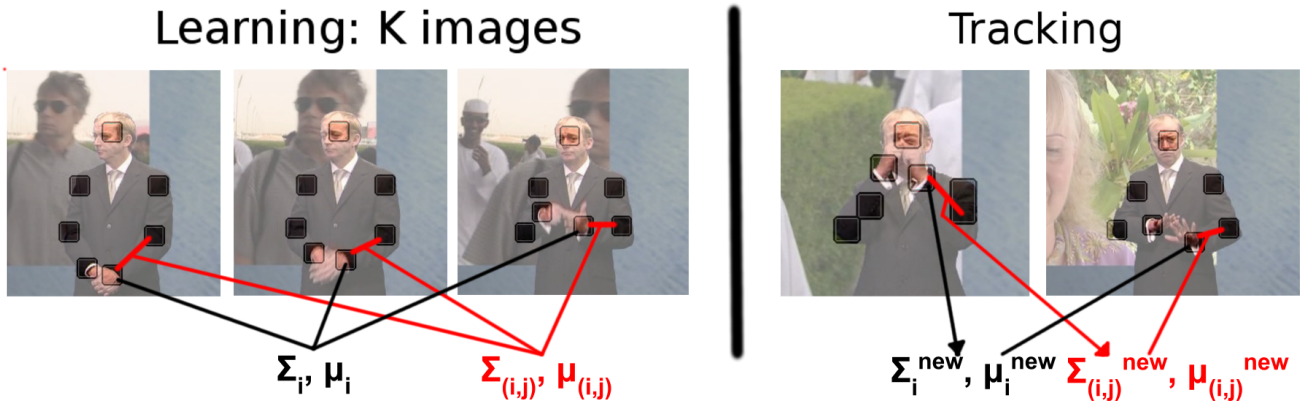



Fig. 2: (best viewed in colour) Visual overview of the proposed part-based model-free tracking system where the patches are indicated with more intense colours. In the learning part (left side of the figure) the appearance parameters are learnt, such a visualisation is provided for part i (the left hand). Similarly the spatial parameters are learned from the relative movement (deformation) of the parts. On the right side of the figure the tracking is performed. The incremental parameter updates, which allow the method to account for (appearance) adaptations, are also indicated in the tracking part.

symbol $\boldsymbol{i}^{(t)}$ denotes a vectorised 2D image[3].

For each frame $\boldsymbol{i}^{(t)}$ in time $t$, we estimate a set of $n$ points with spatial configuration

$$\boldsymbol{l}^{(t)} = [[\boldsymbol{\ell}_1^{(t)}]^T, [\boldsymbol{\ell}_2^{(t)}]^T, \ldots, [\boldsymbol{\ell}_n^{(t)}]^T]^T$$

where $\boldsymbol{\ell}_j^{(t)} = [x_j^{(t)}, y_j^{(t)}]^T, j \in [1, n]$ denote the Cartesian coordinates of the j$^{\text{th}}$ point. For patch shape $(w_j, h_j)$ (patch area $p_a = w_j \cdot h_j$), $\boldsymbol{i}_j \in \mathbb{R}^{p_a}$ declares a vectorised rectangular image patch of height $h_j$ and width $w_j$ defined by the spatial neighbourhood centered around the point $\boldsymbol{\ell}_j$ . Additionally, the relative location of two points of interest is determined as the vector of their spatial difference, i.e. $\boldsymbol{\ell}_j - \boldsymbol{\ell}_k = [x_j - x_k, y_j - y_k]^T$. The $\boldsymbol{\mathcal{I}}$ denotes an identity matrix of appropriate dimensionality.

### *B. Learning*

Given a frame $\boldsymbol{i}^{(t)}$, we model the likelihood $P(\boldsymbol{i}^{(t)}, \boldsymbol{l}^{(t)} | \boldsymbol{A}, \boldsymbol{S})$ where $\boldsymbol{A}, \boldsymbol{S}$ denote the appearance, shape parameters respectively. This likelihood expresses the confidence of observing the part $\boldsymbol{\ell}_j$ in the patch $\boldsymbol{i}_j^{(t)}$ which captures both the appearance variation and the spatial relationship of the points of interest. Following the successful paradigm of pictorial structures [20] we create a tree $G = (V, E)$ where each vertex $V = \{v_1, v_2, ..., v_n\}$ corresponds to a point of interest $j$, while each edge $E$ models the structural constraints between every pair of points that are connected. That tree enables us to use the efficient derivation conducted by Felzenszwalb *et al.*in [22]. Then the likelihood is equivalent to:

$$P(\boldsymbol{i}^{(t)}, \boldsymbol{l}^{(t)} | \boldsymbol{A}, \boldsymbol{S}) = P(\boldsymbol{i}^{(t)} | \boldsymbol{l}^{(t)}, \boldsymbol{A}) P(\boldsymbol{l}^{(t)} | \boldsymbol{S}) = \prod_{j=1}^{n} P(\boldsymbol{i}^{(t)} | \boldsymbol{\ell}_j^{(t)}, \boldsymbol{A}) \prod_{(v_k, v_j) \in E}^{n} P(\boldsymbol{\ell}_j^{(t)}, \boldsymbol{\ell}_k^{(t)} | \boldsymbol{S}) \quad (1)$$

[3]The extension in case of feature extraction method or any other transformation in the images is straightforward.

The aforementioned tree-structure enables us to utilise the generalised distance transforms [22] for maximising the likelihood of Eq. 1. Notice that Eq. 1 depends on two product terms; the first term captures the conditional appearance probability, while the second the spatial configuration probability. Each of the two terms is modelled separately below with the final objective of maximizing the likelihood.

#### Appearance modeling

The appearance of each point $j$ is modelled with a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ ($\boldsymbol{\mu}_j \in \mathbb{R}^{p_a}$, $\boldsymbol{\Sigma}_j \in \mathbb{R}^{(p_a, p_a)}$). The negative logarithm of the $P(\boldsymbol{i}^{(t)} | \boldsymbol{\ell}_j^{(t)}, \boldsymbol{A})$ is minimised instead of the maximisation of the likelihood, results in optimising:

$$\arg\min_{\boldsymbol{i}_j} (\boldsymbol{i}_j - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1} (\boldsymbol{i}_j - \boldsymbol{\mu}_j) \quad (2)$$

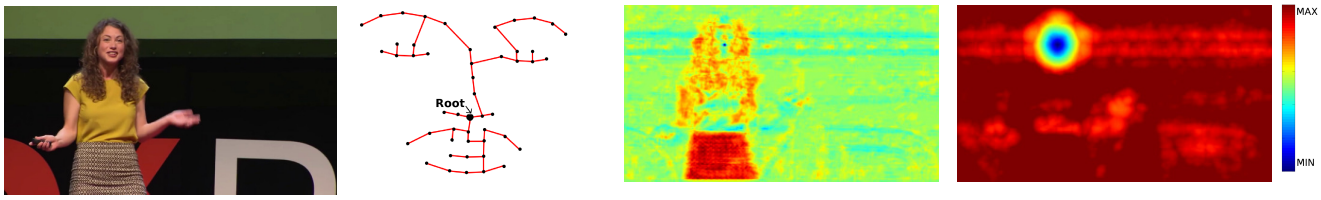
We perform a Singular Value Decomposition (SVD) in every matrix $\boldsymbol{\Sigma}_j$ and maintain the $m$ greatest eigenvalues. Hence, $\boldsymbol{\Sigma}_j \approx \boldsymbol{U}_j \boldsymbol{\mathcal{L}}_j \boldsymbol{U}_j^T$ with $\boldsymbol{U}_j \in \mathbb{R}^{(p_a, m)}$, $\boldsymbol{\mathcal{L}}_j$ a diagonal matrix of shape $m \times m$.

#### Reconstruction error

The model-free context along with the Gaussian distribution assumption, constrains the representational power of $\boldsymbol{\Sigma}_j$ within the learnt subspace. We ameliorate that by enriching the variance formulation. Each patch $\boldsymbol{i}_j^{(t)}$ is modelled as a linear combination of a latent random variable $\boldsymbol{r}_j$ with $\boldsymbol{r}_j \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\mathcal{L}_j})$, the mean appearance $\boldsymbol{\mu}_j$ and a random variable $\boldsymbol{\varepsilon}$ with $\varepsilon \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{\mathcal{I}})$ that incorporates the noise. The formula for $\boldsymbol{i}_j^{(t)}$ is:

$$\boldsymbol{i}_j^{(t)} = \boldsymbol{U}_j \boldsymbol{r}_j + \boldsymbol{\mu}_j + \boldsymbol{\varepsilon} \quad (3)$$

Based on the previous equation, the posterior of $\boldsymbol{i}^{(t)}$ is




(a) Original frame (b) Minimum Spanning Tree (c) Appearance (unary) cost (d) Total cost (root point)

Fig. 3: (Preferably viewed in colour) Visualisation of the tree structure and the unary/total cost for a sample frame (for the 49 facial markup). The heat-maps demonstrate the cost that should be minimised; the preferred (argmin) points are the ones closer to the blue (denoted in the colour range). The unary cost is quite precise in this sample, however the total cost that accounts for the spatial deformations is more accurate (filtering out with very high values all the unrelated coordinates).

modified to:

$$P(\boldsymbol{i}^{(t)}|\boldsymbol{\ell}_j^{(t)},\boldsymbol{A}) = \int_{\boldsymbol{r}_j} P(\boldsymbol{i}^{(t)}|\boldsymbol{r}_j,\boldsymbol{\ell}_j^{(t)},\boldsymbol{A})P(\boldsymbol{r}_j|\boldsymbol{A})d\boldsymbol{v}_j = \mathcal{N}(\boldsymbol{\mu}_j,\boldsymbol{U}_j\boldsymbol{\mathcal{L}}_j\boldsymbol{U}_j^T+\sigma^2\boldsymbol{\mathcal{I}}) \quad (4)$$

Note that the posterior in Eq. 4 is a Gaussian with the same mean as the original one while the variance is augmented with a term $\sigma^2\boldsymbol{\mathcal{I}}$, i.e. $\boldsymbol{\Sigma}_j^{augm} = \boldsymbol{U}_j\boldsymbol{\mathcal{L}}_j\boldsymbol{U}_j^T + \sigma^2\boldsymbol{\mathcal{I}}$. Hence, following the derivation above, it is sufficient to find $argmin_{\boldsymbol{i}_j^{(t)}}(\boldsymbol{i}_j^{(t)}-\boldsymbol{\mu}_j)^T(\boldsymbol{\Sigma}_i^{augm})^{-1}(\boldsymbol{i}_j^{(t)}-\boldsymbol{\mu}_j)$. By applying the Woodbury formula:

$$(\boldsymbol{\Sigma}_j^{augm})^{-1} = \boldsymbol{U}_j(\boldsymbol{\mathcal{L}}_j+\sigma^2\boldsymbol{\mathcal{I}})^{-1}\boldsymbol{U}_j^T + \frac{1}{\sigma^2}(\boldsymbol{\mathcal{I}}-\boldsymbol{U}_j\boldsymbol{U}_j^T) \quad (5)$$

As suggested in the experiment III-C, the initial term of $\boldsymbol{U}_j(\boldsymbol{\mathcal{L}}_j+\sigma^2\boldsymbol{\mathcal{I}})^{-1}\boldsymbol{U}_j^T$ does not contribute significantly, thus in our experiments we consider that $\boldsymbol{\Sigma}_j^{augm} \approx \boldsymbol{\Sigma}_j^{reconst} = \frac{1}{\sigma^2}(\boldsymbol{\mathcal{I}}-\boldsymbol{U}_j\boldsymbol{U}_j^T)$.

**Incremental update**

Since there is no prior knowledge about the point of interest, we need to account for the adaptations of the appearance of $\boldsymbol{i}_j^{(t)}$ over time. When $\mathcal{K}_{new}$ images are available the incremental update of each appearance model can be exact as Ross *et al.*proved in [18], i.e. it is equivalent to training a new appearance model with an augmented set of images $\boldsymbol{i}^{1:(\mathcal{K}_0+\mathcal{K}_{new})}$. The derivation of that update is the following:

Given the previous mean $\boldsymbol{\mu}_j$, the eigenvectors $\boldsymbol{U}_j$, the diagonal matrix $\boldsymbol{\mathcal{L}}_j$ of the previous covariance $\boldsymbol{\Sigma}_j$ and the new data $\boldsymbol{B} = \boldsymbol{i}^{\mathcal{K}_0+1:\mathcal{K}_{new}}$ the goal of the update consists in learning a $\hat{\boldsymbol{\mu}}_j$ and a $\hat{\boldsymbol{U}}_j$, $\hat{\boldsymbol{\mathcal{L}}}_j$ with $\hat{\boldsymbol{\Sigma}}_j = \hat{\boldsymbol{U}}_j\hat{\boldsymbol{\mathcal{L}}}_j\hat{\boldsymbol{U}}_j^T$. The new $\mathcal{K}_{new}$ samples can be described by the components already included in $\boldsymbol{U}_j$ and the components to the orthogonal subspace to $\boldsymbol{U}_j$. Denoting the latter components as $\boldsymbol{V}_j$, we obtain

$$\hat{\boldsymbol{\Sigma}}_j = [\boldsymbol{U}_j\ \boldsymbol{V}_j]\begin{bmatrix}\boldsymbol{\mathcal{L}}_j & \boldsymbol{U}_j^T\boldsymbol{B}\\ \boldsymbol{0} & \boldsymbol{V}_j^T\boldsymbol{B}\end{bmatrix}\begin{bmatrix}\boldsymbol{U}_j^T & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{\mathcal{I}}\end{bmatrix} \overset{SVD}{=} ([\boldsymbol{U}_j\ \boldsymbol{V}_j]\,\tilde{\boldsymbol{U}}_j)\,\tilde{\boldsymbol{\mathcal{L}}}_j\,(\tilde{\boldsymbol{U}}_j^T\begin{bmatrix}\boldsymbol{U}_j^T & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{\mathcal{I}}\end{bmatrix}) \quad (6)$$

Note that for the derivation of the last part of the equation, a SVD was performed in the matrix $R = \begin{bmatrix}\boldsymbol{\mathcal{L}}_j & \boldsymbol{U}_j^T\boldsymbol{B}\\ \boldsymbol{0} & \boldsymbol{V}_j^T\boldsymbol{B}\end{bmatrix} = \tilde{\boldsymbol{U}}_j\,\tilde{\boldsymbol{\mathcal{L}}}_j\,\tilde{\boldsymbol{U}}_j^T$. Hence, the updated terms are $\hat{\boldsymbol{U}}_j = [\boldsymbol{U}_j\ \boldsymbol{V}_j]\,\tilde{\boldsymbol{U}}_j$ and $\hat{\boldsymbol{\mathcal{L}}}_j = \tilde{\boldsymbol{\mathcal{L}}}_j$, while the $\hat{\boldsymbol{\mu}}_j$ is the weighted mean of $\boldsymbol{\mu}_j$ combined with the mean of the new images.

**Spatial Modelling**

For each pair of points that are connected with an edge, we model their relative displacement $(\boldsymbol{\ell}_j^{(t)}-\boldsymbol{\ell}_k^{(t)})$ with a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_{j,k},\boldsymbol{\Sigma}_{j,k})$ where $\boldsymbol{\mu}_{j,k}\in\mathbb{R}^2$, $\boldsymbol{\Sigma}_{j,k}\in\mathbb{R}^{(2,2)}$.

In a similar manner to the appearance modelling, the maximisation of the likelihood $P(\boldsymbol{\ell}_j^{(t)},\boldsymbol{\ell}_k^{(t)}|\boldsymbol{S})$ is equivalent to $\arg\min_{\boldsymbol{\ell}_j^{(t)},\boldsymbol{\ell}_k^{(t)}}(\boldsymbol{\ell}_j^{(t)}-\boldsymbol{\ell}_k^{(t)}-\boldsymbol{\mu}_{j,k})^T\boldsymbol{\Sigma}_{j,k}^{-1}(\boldsymbol{\ell}_j^{(t)}-\boldsymbol{\ell}_k^{(t)}-\boldsymbol{\mu}_{j,k})$. This is a computationally demanding derivation as there are thousands of spatial locations in every frame, however by utilising the efficient derivation of the tree structure, it is computed in linear time in the number of spatial locations.

Even though this formulation was developed for spatial modelling, acknowledge that it additionally models the case of deformation in the case of the vertexes of the graph being points in a single object, e.g. a human face.

**Tree structure**

The tree $G$ is constrained so that every vertex aside of the root has exactly one parent vertex. Along with the mean vector $\boldsymbol{\mu}_{j,k}$ and the covariance matrix $\boldsymbol{\Sigma}_{j,k}$ the tree structure and the edges' weights form the spatial parameters $\boldsymbol{S}$ to be learnt. As demonstrated in [20] the optimal configuration for the edges $E$ is provided by computing the Minimum Spanning Tree (MST), which consitutes a tree with minimum total weights on the edges. To learn the structure and the edges, a complete graph with the vertices $V$ and the edges is initialised and then Kruskal's algorithm is applied to compute the MST.

### C. Inference

**Cost function**

The final cost function as derived from Eq. 1 is:

$$\arg\min_{\boldsymbol{l}^{(t)},\boldsymbol{i}_j^{(t)}}(\sum_{j=1}^{n}(\boldsymbol{i}_j^{(t)}-\boldsymbol{\mu}_j)^T\boldsymbol{\Sigma}_j^{-1}(\boldsymbol{i}_j^{(t)}-\boldsymbol{\mu}_j)+ (\boldsymbol{\ell}_j^{(t)}-\boldsymbol{\ell}_k^{(t)}-\boldsymbol{\mu}_{j,k})^T\boldsymbol{\Sigma}_{j,k}^{-1}(\boldsymbol{\ell}_j^{(t)}-\boldsymbol{\ell}_k^{(t)}-\boldsymbol{\mu}_{j,k})) \quad (7)$$

where $k$ denotes the parent node of each vertex and $\boldsymbol{l}^{(t)}$ the configuration of all the parts. Each vertex contributes

an appearance (unary) cost plus the cost of the deformation (pairwise term) from the nominal displacement from its' parent position.

**Efficient computation**

Due to the tree structure, the overall cost can be computed very efficiently by traversing the tree exactly twice (bottom-up and top-down). Specifically, starting from the cost of the leaf nodes and inversely traversing the tree towards the root vertex, the cost of placing each vertex in every position of the grid is computed by utilising the generalised distance transforms. Summing all these costs in the root vertex, the minimum cost position for the root vertex is determined and based on this, the positions of the children are decided recursively by traversing the tree, hence the complete configuration $\boldsymbol{l}^{(t)}$ is derived. An illustrative figure including the minimum spanning tree, along with the unary cost of the root and the final minimum cost position (for the root) is depicted in Fig. 3.

## III. Experiments

The diverse set of comparisons that was performed is summarised in this section (an overview of the datasets exists in Table I). In sec. III-C an internal evaluation of IPST was performed; sequentially the comparisons with state-of-the-art methods in three different tasks (four datasets employed in total) were developed. The tasks selected were: Deformable face tracking [14], [23] in a model-free structure; body parts tracking by employing the two diverse datasets [24], [25] and deformable animal tracking.

Except for SPOT [4] there are no other state-of-the-art methods suitable for deformable model-free tracking (when the parts are explicitly defined). Alternative choices would be incremental AAMs [26] or the incremental discriminative tracker proposed in [27]. We experimented with AAMs in a model-free framework; we found them unstable, not being able to track the object after few frames, contrary to model-based AAMs which work well. The problem was even more pronounced with methods such as Asthana *et al.* [27] which are unstable without the use of a model for a particular object (e.g. faces).

We included a number of single bounding box trackers in our experimentation. Executing a single object model-free tracker for multiple points was both time-consuming, e.g. calling the tracker 49 times for a 49 mark-up annotation for a single clip, and suboptimal since it fails to capture the spatial relationship among the parts. The latter one was especially evident in the experiments where the different parts interacted with finite degrees of freedom, e.g. in the facial landmark mark-up. Nevertheless, we selected the following strong performing methods: (i) CMT [28], which is a recent keypoint-based model-free tracker that implicitly performs deformable tracking (in the single bounding box case); (ii) FCT [29], which is a fast tracker (used as a baseline); (iii) DSST [5], which is a fast and accurate tracker based on correlation filters; (iv) IVT [18], which is among the most widely used trackers as baseline, while it is one of the generative methods that is incrementally updated in a way similar to ours, (v) STCL [30], which utilises a bayesian framework to learn the spatio-temporal context of the object of interest, hence implicitly accounting for the correlation of 'close enough' parts.

Additionally, we implemented an 'oracle'. Principal Component Analysis (PCA) was applied to the first $\mathcal{K}_0$ training frames; then each new shape (frame) was projected and reconstructed from the PCA after removing the global transformation components. The 'oracle' requires access to the ground-truth landmarks for each frame, which are then projected and reconstructed per part, hence it cannot be applied to a new dataset, it is only added for denoting the theoretical upper bound of our method.

The cumulative error distribution (CED) plot is utilised as a comparison metric in the body tracking and the deformable (facial, animal) tracking experiments. Similarly to [25] the x-axis denotes the euclidean distance of the tracked point from the ground-truth, while the y-axis the proportion of images with less than this error. Unless differently mentioned, CED curves measure the image proportion in the y-axis and the (normalised) point-to-point distance in the x-axis. Additional error metrics for every experiment are deferred to the supplementary material. A qualitative result can be found in Fig. 5, visually indicating the strengths of IPST; SPOT is chosen as the main state-of-the-art method in part-based tracking; DSST is randomly chosen among the single bounding box model-free trackers; in the supplementary material additional results are plotted[4].

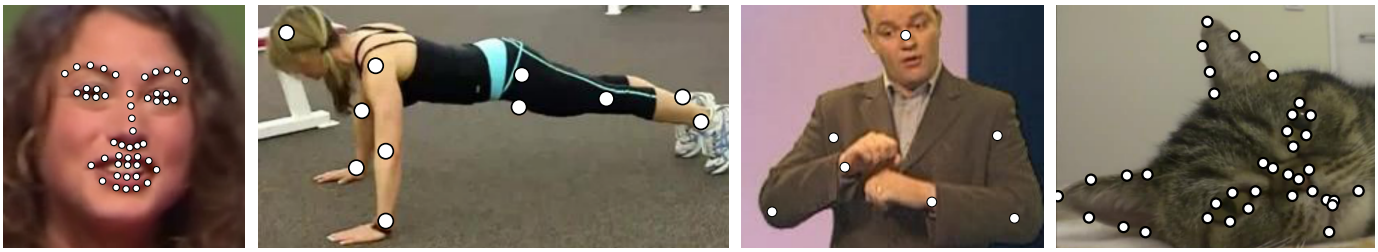
### A. Implementation details

The implementation was conducted inside the Menpo project [31]. In this work the first frame was initialised from the ground-truth; the next four were tracked with a simple template matching per landmark (simple correlation filter per part), i.e. $\mathcal{K}_0 = 5$. We experimentally noticed that the first 4, 5 principal components of each part suffice for tracking. To avoid overflowing values in the shape covariance matrices in case the part movement is negligible in the first $\mathcal{K}_0$ frames, we restrict the inverse covariance values within a pre-defined interval. The sparse HOG features [15] were applied as the feature extraction method. For all the experiments we kept all the parameters of the algorithms unchanged. Regarding the SPOT tracker, we made our own implementation but also tested the code of the authors. We always reported the best performing implementation for each specific video. Any direct comparison with pre-trained methods utilising object domain knowledge might differ from this work's setup, since in model-free tracking framework there is no a priori knowledge of the objects; the trackers have only access to the first $\mathcal{K}_0$ frames of each video.

### B. Computational cost

All the methods were called for the first 100 frames of a video in 300vW [23] (resolution of $1280 \cdot 720$). The machine

[4]A tracking video is provided in https://youtu.be/gCudSfSkYmU. Additional information about it can be found in the supplementary material.



(a) Sec. III-D (b) Sec. III-E (Penn) (c) Sec. III-E (BBC pose) (d) Sec. III-F

Fig. 4: Exemplar frames along with the points of interest for the datasets utilised.

TABLE I: A summary of the diverse datasets used in the experiments. The sign of '#' abbreviates the 'number of'. Columns 3-5 provide statistics only for the videos annotated, i.e. those reported in the experiments. Column 6 mentions the section with the respective experimental setup. In Fig. 4 a visual illustration of exemplar frames for each dataset is provided.

| *Dataset* | *Citation* | *# videos* | *# frames* | *# points* | *Sec.* |
|---|---|---|---|---|---|
| 300 vW | [23] | 64 | 123405 | 49 | III-D |
| BBC pose | [25] | 10 | 1993 | 7 | III-E |
| Penn Action | [24] | 2297 | 162158 | 7-13 | III-E |
| Cat faces | - | 10 | 5861 | 38 | III-F |

employed included an i7 processor, 3.6 GHz, while no further optimisation was performed for any of the methods. The indicative times in seconds are reported in Tab.II. Our implementation was not optimised for the computational cost, hence our python code could be further improved. Nevertheless, it is faster than the SPOT implementation and as indicated in the quantitative results, it outperforms all the compared methods by a large margin in different datasets.

It should be noted that even with CMT which is a recent tracker with a fast implementation (C++), the part-based execution requires over 22 hours. Therefore, only trackers which require less than 3 seconds per frame (for all the 49 parts) were compared.

### C. Self evaluation

Three self evaluation experiments were conducted for quantitatively assessing the proposed method's performance with different parameters. In all cases, the validation was performed in category 1 of 300 Videos in-the-Wild dataset (300vW) [23] that includes over 60000 frames.

**Effect of Sigma and incremental update**: In this experiment we assessed i) the influence of different covariances ($\mathbf{\Sigma}_i^{augm}$ and $\mathbf{\Sigma}_i^{reconst}$), ii) the influence of incremental update.

Three alternatives were considered: a) disable the incremental update, alleged no-update version, b) use the full $\mathbf{\Sigma}_i^{augm}$ as proposed in sec II, c) use only the reconstruction error for the appearance, i.e. the $\mathbf{\Sigma}_i^{reconst}$.

The cumulative curve in Fig. 10 demonstrates the benefit of utilising the incremental update for our learnt subspaces while tracking. Additionally, the difference between $\mathbf{\Sigma}_i^{augm}$ and $\mathbf{\Sigma}_i^{reconst}$ was minor (also visually verified in different experiments), henceforth in the subsequent experiments only the reconstruction error ($\mathbf{\Sigma}_i^{reconst}$) was computed.

**Influence of patch size**: We investigated the sensitivity of IPST to the patch size selection. Even with our efficient vectorised computation (sec. II-C) the patch size has an influence in the computational complexity. Thus, we explored the effect of the patch size in the performance.

The following three options were considered: a) patch size 12, b) patch size 20, c) patch size 24. The CED plot is illustrated in Fig. 7a. The difference between the patch size of 24 and 20 was not substantial, however we chose the patch size 24 for our experiments as it had a marginal improvement. The performance of a much smaller patch size (12) was decreased, i.e. the difference from patch size 24 was substantial.

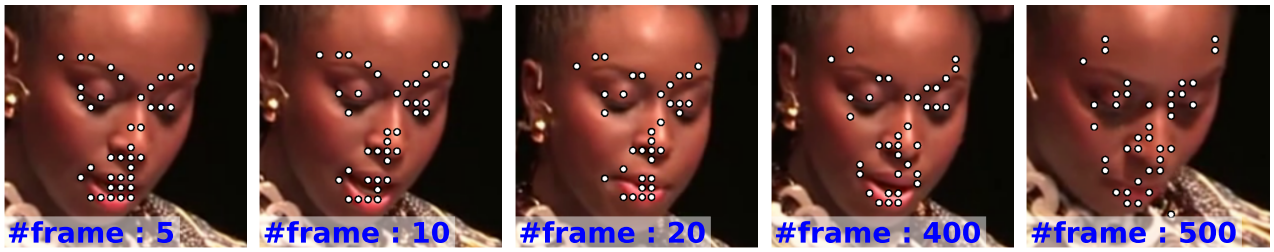
**Rapid motion**: Even though the diverse experiments demonstrated the effectiveness of IPST versus existing methods, we experimented with the case of faster and abrupt changes. That effect was simulated by skipping every second frame of the original database, i.e. assuming that each clip included the sequence $\{2 \cdot n\}$ frames where $n \in \{1, 2, \ldots, \frac{length(clip)}{2}\}$.

We executed IPST in the clips with skipped frames and visualised the CED plot in Fig. 7b. Even though there was a minor decrease in the performance, we noticed that IPST was robust to more rapid motion.

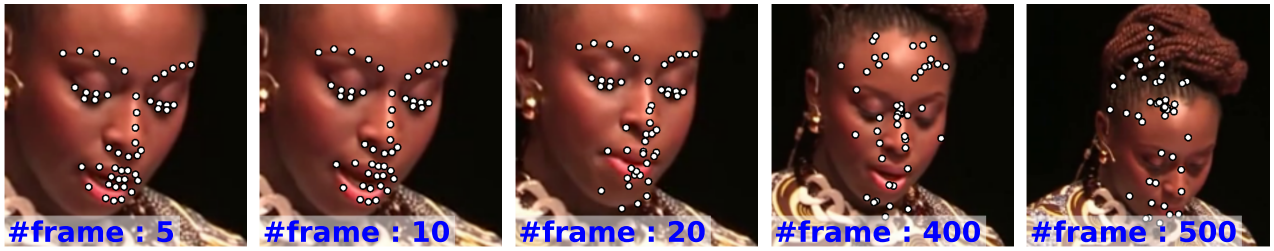
In case prior knowledge about the database was available (e.g. rapid movement, frequent occlusions) more elaborate engineering tricks could be employed. For instance, we could include the forgetting factor [18] that applies a weighting scheme to favour more recent samples. Additionally, the reconstruction error could be used to dictate when to perform updates. However, the simple proposed system was quite robust; further analysis on such tricks is left as future work.

### D. Deformable face tracking

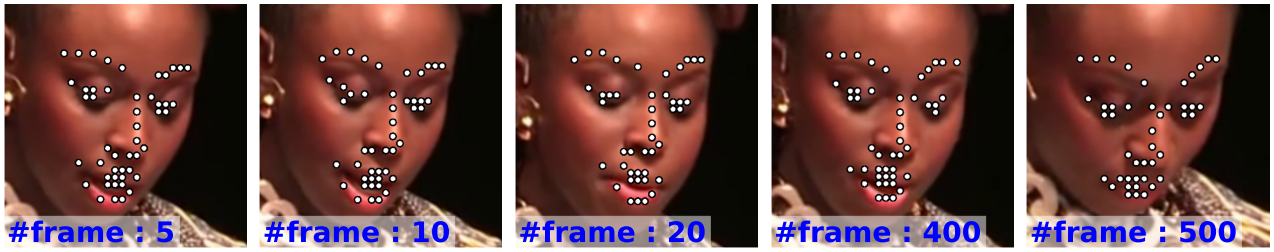
In this experiment, the points of interest were the sparse facial landmark points in the 300 Videos in-the-Wild dataset [23]. This dataset includes 114 videos (approx. 1500 frames each), organised in 3 categories. Each frame contains a single human face and is annotated with 68 points, while the videos comprise of a wide variety of facial poses and expressions. We considered each point of the 49 mark-up as a tracking target and tracked all the frames.




(a) SPOT (facial tracking)




(b) Tracker DSST (facial tracking)




(c) (Proposed) IPST (facial tracking)


(d) SPOT (body tracking)


(e) Tracker DSST (body tracking)

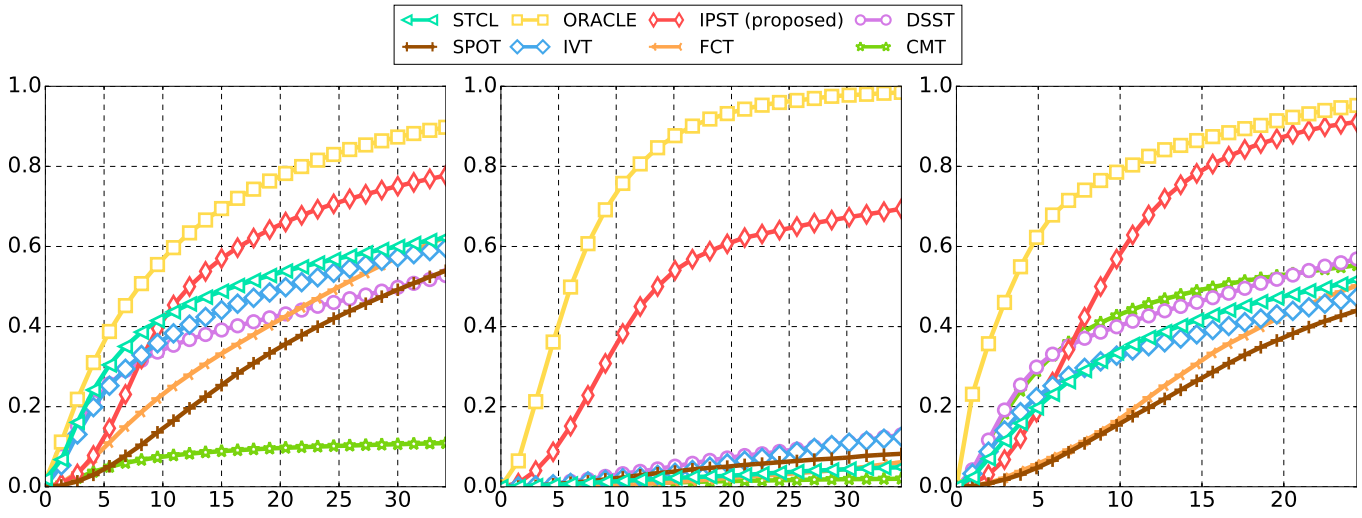

(f) (Proposed) IPST (body tracking)

Fig. 5: (Preferably viewed in colour) Indicative tracked frames from the facial tracking experiment and the body tracking experiment (Penn Action dataset in III-E). Evidently, the single bounding box tracker fails to capture the spatial relationship among the patches; different patches are tracked in random parts, i.e. completely losing the shape information.

| CMT | DSST | FCT | IPST | IVT | SPOT | STCL |
|---|---|---|---|---|---|---|
| 2.4 | **1.3** | **0.9** | **1.3** | 1.5 | 2.2 | 1.4 |

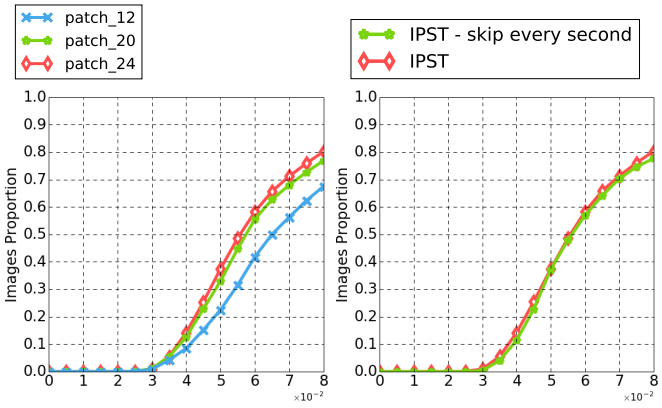
Colouring denotes the methods' ranking: first second third

TABLE II: The computational cost (reported in seconds) of the compared methods. The 3 fastest methods are highlighted.




(a) Sec. III-E (Penn) (b) Sec. III-E (BBC pose) (c) Sec. III-F

Fig. 6: (Preferably viewed in colour) CED plots comparing IPST with the prior art in different scenarios. The *x-axis* represents the point-to-point error per frame, while the *y-axis* the proportion of frames up to an error. (b), (c) The comparison with the prior art on body tracking in two diverse datasets. (d) The animal tracking experiment portrays the ad-hoc utilities of the IPST for tracking deformable objects when no prior annotations exist. The proposed method outperforms the compared methods.




(a) Influence of patch size (b) Rapid motion

Fig. 7: (Preferably viewed in colour) Self evaluation related CED plots. (a) The CED plot reflecting the influence of different patch sizes. The x-axis depicts the normalised point-to-point error, while the y-axis the proportion of images with less than a specified x error. (b) The CED plot for rapid motion case.

Following the deformable literature on faces [14], [32], the standard cumulative error distribution (CED) plots with normalised point-to-point error was selected for the quantitative evaluation. The CED curves were produced based on the ones in Chrysos *et al.* [14] for 49 landmark points. The error metric chosen was the mean Euclidean displacement of the 49 points, normalised by the length of the ground truth's bounding box diagonal[5]. The cut-off error was set to 0.08; above that threshold it was considered failure to localise the landmarks.

We visualise the CED plots of all three categories in Fig. 8. In category 1 (Fig. 8a) the proposed method managed to localise approximately 40% of the landmarks (across all images) with error less than 0.05 (mediocre errors), while it is noticeable that the rest methods did not manage to reach this point even in 0.08 error. IPST largely outperformed the rest compared methods. In category 2 (Fig. 8b) IPST still outperformed the compared methods, however the performance of all methods was slightly improved. This was credited to the robustness of modern trackers to illumination changes (which is the challenge of category 2). In this category the single bounding box trackers performed considerably worse than methods that consider explicitly the relationship among the parts. Even with trackers that localised the parts in the first few frames accurately, the high correlation of the points in the human face dominated the localisation ability, hence they drifted within the first few hundred frames and tracked 'random' patches of the image. Lastly, category 3 (Fig. 8c), which is the most challenging in 300VW, validated the results of the previous two categories. Even though the performance of all methods deteriorated, IPST outperformed the compared methods. The decreased performance of category 3 was attributed to the self-occlusions of the face. Only the first few frames were observed and when those points were occluded, the model-free methods could not learn the appearance of the occluded part, hence they tracked the patch that was causing the occlusion in the first frames; please see Fig. 9 for an illustration of the phenomenon.

To sum up, the spatial modelling contributed to the successful tracking of the facial parts in all three categories. The discriminative training of SPOT with only few positive samples, made it very challenging to disambiguate between similar

[5]As reported in [14] this metric is relatively invariant to large pose angles, while it is mathematically expressed as $(\sqrt{width^2 + height^2})$. $height$ and $width$ correspond to the vertical and horizontal difference of the boundary landmark points respectively.

parts, e.g. the eyes. Additionally, in SPOT the deformation cost parameters are constant instead of being learnt from data, hence it got trapped easily in local minima.

### E. Deformable body parts tracking

The popular datasets of Charles *et al.* [25] and Zhang *et al.* [24] were selected for studying the movement of fiducial body parts.

**BBC pose dataset:** The extended BBC pose database [25] includes 92 videos of news in sign language; each video includes several thousands frames. For each frame, 7 points of the presenter's body are annotated. In this experiment we assessed the methods' performance only in the manually annotated frames, which are provided for 10 videos (200 samples per video are manually annotated). BBC pose consists a very challenging benchmark due to the abrupt background changes, e.g. complete change of the background in two consequtive frames.

The CED plot of Fig. 6b demonstrates that the performance of IPST outperforms significantly the compared methods. Their poor performance can be attributed to the fast background changes; the background can completely change from a frame to the next, resulting in complete drifting of the compared methods.

**Penn Action dataset:** The Penn Action dataset [24] was introduced for action understanding and includes 2326 short clips; each clip includes a single action performed by a human; up to 13 points in the body are annotated per frame. Through the visibility masks provided we tracked only the visible points per clip, *i.e.*only the points visible in the first 10 frames were considered. Considering the amount of videos that this dataset contains along with the fact that in every video the bodies annotated are in a different pose from the rest of the videos, this makes it one of the most challenging benchmarks for comparing methods for deformable tracking. The difference in the results (Fig. 6a) from BBC can be attributed to the shorter length of the clip (approx. 70 frames), which is very short in comparison to the longer clips of BBC. Nevertheless, IPST did handle the diverse actions and appearance/spatial modelling cases and outperformed the compared methods. In addition, in the supplementary material we study how the curve is modified in case we consider only the clips with a minimum of 50 frames.

### F. Deformable animal tracking

Animals' faces express a larger degree of shape and appearance variation than human faces. To the best of our knowledge, there is no tracking dataset with sparse shapes on animals' bodies, hence 10 videos with 38 markup facial points in cat faces were annotated semi-automatically per frame. Particularly, utilising the 350 fittings of Sagonas *et al.* [33] a patch-based AAM was built [31], [34], then a video-specific method ( [13]) was used for the refinement of the points. The erroneous annotations were excluded by two experts. This annotation process is only required for obtaining the ground-truth shapes and not for the actual methods to track.

This experiment illustrated the ad-hoc utility of IPST in the model-free tracking framework. The lack of such a markup for animal tracking required the laborious process of annotating plenty of frames; building elaborate models [34]. However, with IPST only the first few annotations are required and then it tracks the rest of the frames automatically. This provides a satisfying initialisation and then possibly a refinement with some video-specific trained model would provide very accurate results, as can be verified in Fig. 6 with the CED plot of this experiment. A qualitative comparison is provided in Fig. 11 for different frames of a video of the category. As aforementioned, the methods of single bounding box trackers do fail after the first couple of frames, since they do not capture the spatial relationships.

To further demonstrate the merits of IPST in an ad-hoc case, we trained a model-based system using MDNET [8] (state-of-the-art tracker) + parametric SDM with 2000 collected images of cats. The collection of these images even with a semi-automatic method is quite costly (please refer to the supplementary material for a further analysis). In order not to clutter the results, we present in Fig. 10 the comparison of IPST with a model-based approach. MDNET + SDM is a state-of-the-art approach, however IPST still is more robust to the wide range of deformations of animals' bodies.

## IV. Conclusion

In this work we introduced IPST, a method for model-free part-based object tracking. We model the appearance and the deformation of each part with multivariate Gaussian distributions; we update those incrementally based on the tracked frames to account for object adaptations. The thorough assessment in different tasks, i.e. in facial landmark tracking, body parts tracking and animal tracking experimentally demonstrated that IPST outperforms the prior art. Any custom deformable shape could be defined and the proposed method can track that object(s) given only the first few annotated frames. In the future we plan to perform an in-depth study of the frames in which incremental update is required in order to avoid cases of updating the subspaces during an ephemeral occlusion. Another line of research, is the direct utilisation of the learnt subspace from the appearance and spatial models by a more elaborate model-based method for part refinement, e.g. an AAM.

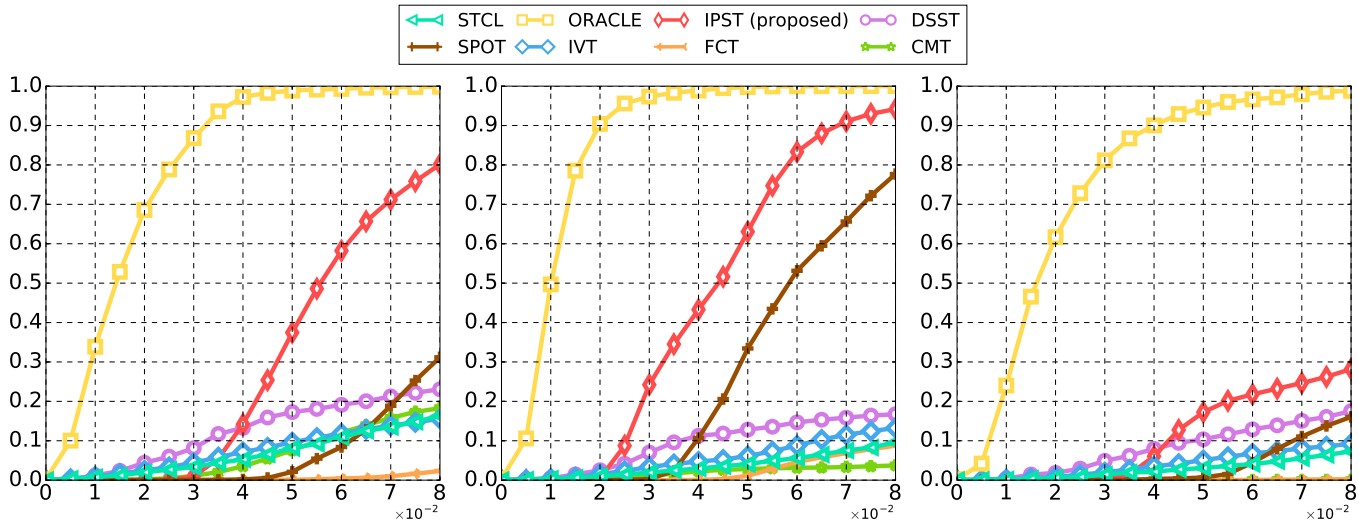
## V. Acknowledgements

The work of Grigorios Chrysos has been funded by an Imperial College DTA. The work of Stefanos Zafeiriou has been partially funded by the FiDiPro program of Tekes (project number: 1849/31/2015), as well as the EPSRC project EP/N007743/1 (FACER2VM).

## References

[1] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Čehovin, G. Fernandez, T. Vojir, G. Häger, G. Nebehay *et al.*, "The visual object tracking vot2015 challenge results," in *IEEE Proceedings of International Conference on Computer Vision Workshops (ICCV'W)*, Dec 2015. 1

[2] Z. Chen, Z. Hong, and D. Tao, "An experimental survey on correlation filter-based tracking," *arXiv preprint arXiv:1509.05520*, 2015. 1




(a) Category 1 (b) Category 2 (c) Category 3

Fig. 8: (Preferably viewed in colour) CED plots comparing IPST with the prior art in deformable face tracking.

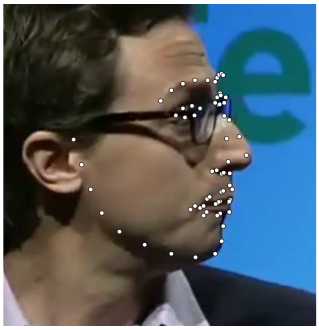
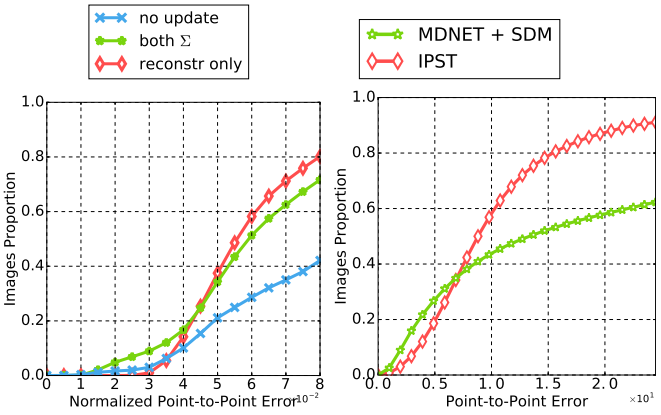


Fig. 9: (Preferably viewed in colour) Profile face from category 3 of 300VW. Due to the self-occluded left part of the face, the position of the landmarks does not correspond to their semantic appearance. If initialised/incrementally updated from this frame, the trackers could drift, which explains the deteriorated performance of category 3 CED plots.




(a) Self evaluation (Sec. III-C) (b) Model-based comparison

Fig. 10: (Preferably viewed in colour) CED plots for (a) the self evaluation and (b) the comparison with a state-of-the-art model based system for the animal tracking experiment (Sec.III-F). In (a) the benefit of applying an incremental update (versus the 'no update') was experimentally verified, while the two versions with $\mathbf{\Sigma}_i^{augm}$ (green line) and $\mathbf{\Sigma}_i^{reconst}$ (red line) were compared.


[3] R. Yao, Q. Shi, C. Shen, Y. Zhang, and A. Hengel, "Part-based visual tracking with online latent structural learning," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 2363–2370. 1, 2

[4] L. Zhang and L. van der Maaten, "Preserving structure in model-free tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, vol. 36, no. 4, pp. 756–769, 2014. 1, 2, 5

[5] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proceedings of British Machine Vision Conference (BMVC)*, 2014. 1, 5

[6] M. Danelljan, G. Häger, F. Shahbaz Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *IEEE Proceedings of International Conference on Computer Vision (ICCV)*, 2015, pp. 4310–4318. 1

[7] R. Tao, E. Gavves, and A. W. Smeulders, "Siamese instance search for tracking," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016. 1

[8] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 9

[9] K. Zhang, Q. Liu, Y. Wu, and M.-H. Yang, "Robust visual tracking via convolutional networks," *arXiv preprint arXiv:1501.04505*, 2015. 1

[10] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1. IEEE, 2006, pp. 798–805. 1

[11] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *IEEE International Conference on Pattern Recognition (ICPR)*. IEEE, 2010, pp. 2756–2759. 1

[12] X. Wang, M. Valstar, B. Martinez, M. Haris Khan, and T. Pridmore, "Tric-track: Tracking by regression with incrementally learned cascades," in *IEEE Proceedings of International Conference on Computer Vision (ICCV)*, 2015, pp. 4337–4345. 1

[13] G. G. Chrysos, E. Antonakos, S. Zafeiriou, and P. Snape, "Offline deformable face tracking in arbitrary videos," in *IEEE Proceedings of International Conference on Computer Vision, 300 Videos in the Wild (300-VW): Facial Landmark Tracking in-the-Wild Challenge & Workshop (ICCV-W)*, 2015, pp. 1–9. 1, 9

[14] G. G. Chrysos, E. Antonakos, P. Snape, A. Asthana, and S. Zafeiriou, "A comprehensive performance evaluation of deformable face tracking "in-the-wild"," *International Journal of Computer Vision (IJCV)*, 2017. 1, 5, 8

[15] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, vol. 32, no. 9, pp. 1627–1645, 2010. 1, 2, 5

[16] E. Antonakos, J. Alabort-i Medina, and S. Zafeiriou, "Active pictorial structures," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 5435–5444. 2

[17] Y. Zhou, E. Antonakos, J. A. i medina, A. Roussos, and S. Zafeiriou, "Estimating correspondences of deformable objects "in-the-wild"," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2016. 2

[18] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision (IJCV)*, vol. 77, no. 1-3, pp. 125–141, 2008. 2, 4, 5, 6

[19] S. Liwicki, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "Euler principal component analysis," *International Journal of Computer Vision (IJCV)*, vol. 101, no. 3, pp. 498–518, 2013. 2

[20] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *International Journal of Computer Vision (IJCV)*, vol. 61, no. 1, pp. 55–79, 2005. 2, 3, 4

[21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 886–893. 2

[22] P. Felzenszwalb and D. Huttenlocher, "Distance transforms of sampled functions," Cornell University, Tech. Rep., 2004. 3

[23] J. Shen, S. Zafeiriou, G. Chrysos, J. Kossaifi, G. Tzimiropoulos, and M. Pantic, "The first facial landmark tracking in-the-wild challenge: Benchmark and results," in *IEEE Proceedings of International Conference on Computer Vision, 300 Videos in the Wild (300-VW): Facial Landmark Tracking in-the-Wild Challenge & Workshop (ICCV-W)*, December 2015. 5, 6

[24] W. Zhang, M. Zhu, and K. Derpanis, "From actemes to action: A strongly-supervised representation for detailed action understanding," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 2248–2255. 5, 6, 9

[25] J. Charles, T. Pfister, M. Everingham, and A. Zisserman, "Automatic and efficient human pose estimation for sign language videos," *International Journal of Computer Vision (IJCV)*, vol. 110, no. 1, pp. 70–90, 2014. 5, 6, 9

[26] J. Sung and D. Kim, "Adaptive active appearance model with incremental learning," *Pattern recognition letters*, vol. 30, no. 4, pp. 359–367, 2009. 5

[27] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic, "Incremental face alignment in the wild," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1859–1866. 5

[28] G. Nebehay and R. Pflugfelder, "Clustering of Static-Adaptive correspondences for deformable object tracking," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015. 5

[29] K. Zhang, L. Zhang, and M.-H. Yang, "Fast compressive tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, vol. 36, no. 10, pp. 2002–2015, 2014. 5

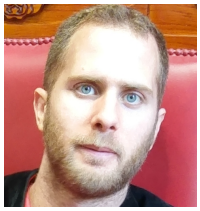
[30] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang, "Fast visual tracking via dense spatio-temporal context learning," in *Proceedings of European Conference on Computer Vision (ECCV)*, 2014, pp. 127–141. 5

[31] J. Alabort-i-Medina, E. Antonakos, J. Booth, P. Snape, and S. Zafeiriou, "Menpo: A comprehensive platform for parametric image alignment and visual deformable models," in *Proceedings of ACM International Conference on Multimedia (ACM'MM)*. ACM, 2014, pp. 679–682. 5, 9

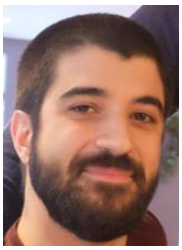
[32] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: Database and results," in *Image and Vision Computing*, 2015. 8

[33] C. Sagonas, Y. Panagakis, S. Zafeiriou, and M. Pantic, "Robust statistical frontalization of human and animal faces," *International Journal of Computer Vision (IJCV)*, pp. 1–22, 2016. 9

[34] G. Tzimiropoulos and M. Pantic, "Gauss-newton deformable part models for face alignment in-the-wild," in *IEEE Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014. 9

**Grigorios G. Chrysos** is a PhD student in iBUG group working with Stefanos Zafeiriou. He in his third year working towards a PhD degree, while previously, he graduated from National Technical University of Athens (2014). He has since been working in the field of Computer Vision and statistical Machine Learning, while he has published his work on deformable models in prestigious journals. He has co-organised workshops for deformable models, e.g. 2D/3D facial landmark tracking, in CVPR/ICCV.



**Epameinondas Antonakos** is an Applied Scientist at Amazon Development Center in Berlin, Germany since February 2017. During 2012-2017, he did his Ph.D. at the Department of Computing, Imperial College London, U.K. as part of the Intelligent Behaviour Understanding Group (iBUG) under the supervision of Dr. Stefanos Zafeiriou. He received his Diploma degree in Electrical and Computer Engineering from the National Technical University of Athens, Greece, in 2011. His research interests lie in the field of Computer Vision and Machine Learning. He has published his Ph.D. work in top-tier conference and journal papers, focusing on 2D and 3D deformable models for fitting and tracking of the human face.

**Stefanos Zafeiriou** (M09) is currently a Reader in Machine Learning and Computer Vision with the Department of Computing, Imperial College London, London, U.K, and a Distinguishing Research Fellow with University of Oulu under Finish Distinguishing Professor Programme. He was a recipient of the Prestigious Junior Research Fellowships from Imperial College London in 2011 to start his own independent research group. He was the recipient of the Presidents Medal for Excellence in Research Supervision for 2016. He currently serves as an Associate Editor of the IEEE Transactions on Affective Computing and Computer Vision and Image Understanding journal. He has been a Guest Editor of over six journal special issues and co-organised over 13 workshops/special sessions on specialised computer vision topics in top venues, such as CVPR/FG/ICCV/ECCV. He has co-authored over 55 journal papers mainly on novel statistical machine learning methodologies applied to computer vision problems, such as 2-D/3-D face analysis, deformable object fitting and tracking, shape from shading, and human behaviour analysis, published in the most prestigious journals in his field of research, such as the IEEE T-PAMI, the International Journal of Computer Vision, the IEEE T-IP, the IEEE T-NNLS, the IEEE T-VCG, and the IEEE T-IFS, and many papers in top conferences, such as CVPR, ICCV, ECCV, ICML. He has more than 4500 citations to his work, h-index 36. He was the General Chair of BMVC 2017.

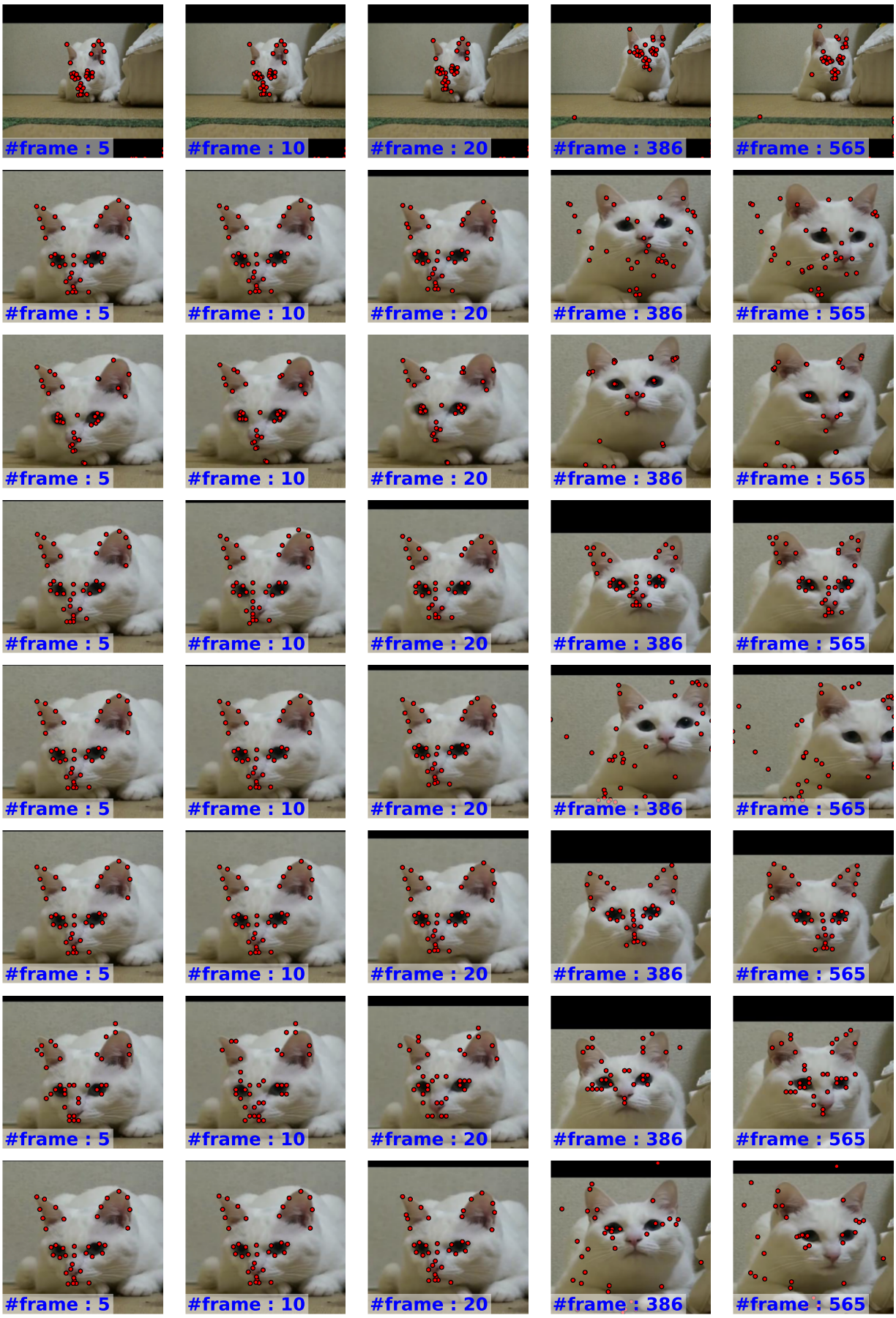



Fig. 11: (Preferably viewed in colour) Indicative tracked frames from the task of animal tracking. The results are in alphabetical order, i.e. : $1^{st}$ row: CMT, $2^{nd}$ row: DSST, $3^{rd}$ row: FCT, $4^{th}$ row: IPST, $5^{th}$ row: IVT, $6^{th}$ row: ORACLE, $7^{th}$ row: SPOT, $8^{th}$ row: STCL.