# Robust visual tracking using template anchors

ehovin, Luka; Leonardis, Ales; Kristan, Matej

*DOI:*
10.1109/WACV.2016.7477570



*Document Version*
Peer reviewed version

*Citation for published version (Harvard):*
ehovin, L, Leonardis, A & Kristan, M 2016, Robust visual tracking using template anchors. in WACV 2016: IEEE Winter Conference on Applications of Computer Vision. IEEE Computer Society Press, 2016 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY, United States, 7/03/16. https://doi.org/10.1109/WACV.2016.7477570

# Visual Tracking Using Anchor Templates

Luka Čehovin, Aleš Leonardis, and Matej Kristan
Faculty of Computer and Information Science, University of Ljubljana, Slovenia
Večna pot 113, Ljubljana, Slovenia
{luka.cehovin,ales.leonardis,matej.kristan}@fri.uni-lj.si

## Abstract

*Deformable part models exhibit excellent performance in tracking non-rigidly deforming targets, but are usually outperformed by holistic models when the target does not deform or in the presence of uncertain visual data. The reason is that part-based models require estimation of a larger number of parameters compared to holistic models and since the updating process is self-supervised, the errors in parameter estimation are amplified with time, leading to a faster accuracy reduction than in holistic models. On the other hand, the robustness of part-based trackers is generally greater than in holistic trackers. We address the problem of self-supervised estimation of a large number of parameters by introducing controlled graduation in estimation of the free parameters. We propose decomposing the visual model into several sub-models, each describing the target at a different level of detail. The sub-models interact during target localization and, depending on the visual uncertainty, serve for cross-sub-model supervised updating. A new tracker is proposed based on this model which exhibits the qualities of part-based as well as holistic models. The tracker is tested on the highly-challenging VOT2013 and VOT2014 benchmarks, outperforming the state-of-the-art.*

## 1. Introduction

Visual object tracking is a research topic with applications in various fields including surveillance, activity recognition, sport analysis and human-computer interaction. The application diversity has resulted in numerous tracking approaches, many of which have been recently compared in papers like [29, 24], and most notably, within the VOT challenges [16, 17]. The results of these comparisons show excellent performance of the holistic models [1, 11, 34, 12, 7], which model the target appearance as a single monolithic representation. These models tend to address well the appearance changes due to illumination or blur, but poorly handle structural appearance changes caused by non-rigid deformations or partial self-occlusions. In the presence of these types of appearance changes, the holistic models start adapting to the background, which results in drifting and eventual failure. The geometrical changes are conceptually better addressed by part-based models [32, 19, 6, 25, 31, 33]. In fact, the recent benchmarks [16, 17], indicate significant robustness of part-based visual models, especially when tracking non-rigid objects.

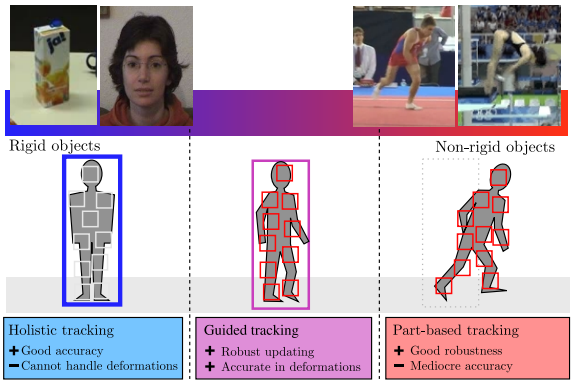


Figure 1. Illustration of our tracking concept that combines holistic visual model and a part-based visual model by switching between them.

Since part-based models are a generalization of holistic models, one might expect equal or better overall performance compared to holistic models regardless whether the target is deforming or not. However, as shown in [16, 17], part-based visual models tend to achieve lower accuracy, especially with rigid targets. We believe that the main reason for this performance drop in the part-based models can be explained by the number of parameters estimated and the uncertainty/noise in the visual data. In the presence of noisy visual data, a holistic visual model that estimates only translation and size, will typically better estimate the position of a rigid object due to the constrained parameterization. The part-based models, on the other hand, deform freely to match the visual data and account for outdated or occluded parts. The uncertainty of the visual data thus introduces potentially small errors in the large number of parameters to be estimated, leading to poor position estimate. Even if

the target is deforming non-rigidly, a low-parameter holistic model might lead to a smaller position error in a short run than the part-based models that would over-fit the uncertain visual data. Still, in the presence of low visual uncertainty, the deformable part models typically outperform the holistic models.

Our main contribution in this paper is introduction of graduated flexibility in parameter estimation of deformable parts trackers. A new visual model composed of several interacting types of visual sub-models that primarily differ in the level of detail by which they describe the target is proposed. The level of detail varies in the type of features used, the number of parameters estimated by each model and the aggressiveness of the adaptation. The sub-models mutually interact in localization and cross-sub-model updates by accounting for the potential uncertainty of the visual information. This makes the visual model shift between purely holistic and part-based behavior, depending on the visual uncertainty (Figure 1). To the best of our knowledge this kind of self-constrained graduated estimation of free parameters has not been proposed before. We have implemented a new visual tracker according to the proposed that achieves a state-of-the-art performance on two recent, highly challenging, visual object tracking benchmarks VOT2013 [16] and VOT2014 [17].

The reminder of the paper is organized as follows: Section 2 positions our contributions against the related work, Section 3 describes the new visual model and its integration into the proposed tracker. Experimental evaluation and results are presented and discussed in Section 4, followed by a conclusion in Section 5.

## 2. Relation to related work

Several works have attempted information fusion for robust visual tracking. Santner et al. [23] proposed running an online discriminative tracker in parallel with motion prediction from a dense optical flow and NCC detection. Trackers that do not address scale change are connected into a cascade and only the model in the discriminative tracker is updated. Badrinarayanan et al. [2] proposed running in parallel two particle filters with different visual models. The two trackers interact by influencing resampling in each tracker. This approach was generalized by Kwon and Lee [18] who proposed a unified framework for a Monte-Carlo-based integration of several holistic visual models in a recursive Bayes filter. In contrast, our work combines holistic visual models with part-based model and therefore addresses scale and aspect changes as well as non-rigid deformations. The visual sub-models are selectively updated depending on the visual certainty of individual models and do not require computationally-intensive sampling. A step towards part-based representation was made by [4, 35] where a SIFT keypoint constellation is combined with a holistic color visual model to better address partial occlusions, however, these methods use rigid constellations of keypoints that are unsuitable for non-rigid objects.

Various part-set models have been increasingly used over the last decade to accommodate geometric deformations of the targets. These models range from rigid-constrained constellations proposed by Vojir et al. [27] to very weakly center-constrained constellations, proposed by Kwon et al. [19]. The rigid constraints in [27] increase robustness during partial occlusions, but suffer from the same deficiencies in tracking deformable targets as the holistic models. Relaxing the constellation constraints increases robustness during object deformations, but also results in high-dimensional optimization problems with many local minima, which are difficult to solve, even for manually initialized set of parts [6]. To reduce this complexity, the topology is usually simplified to a star-based constellation [10, 32, 19]. On the other hand, Čehovin et al. [25] have shown that improved robustness is achieved by densely-connected topology and efficient stochastic optimization. Many part-based models include aggressive updates by removing outdated parts and replacing them with new ones. Kwon and Lee [19] allocate parts from color posterior computed from foreground/background histograms and Čehovin et al. [25] proposed using several features like color and motion. Godec et al. [10] and Duffner and Garcia [8] sample parts from a segmentation mask. The performance of these approaches degrades severely whenever the histogram backprojection or segmentation fail. Our work differs from these approaches in the extension of the update process in which holistic models directly influence part allocation by region proposal, depending on their certainty. Whenever the holistic models are uncertain, the parts are allocated by self-supervision by the part-based model. Our part-based model also uses a densely-connected constellation, but does not require computationally intense stochastic optimization.

The problem of drifting due to continuous visual model updates has been explored in long-term visual tracking by Kalal et al. [13] and Pernici et al. [21]. The common strategy is to model the appearance by a large set of object instances and introduce the instances into the model very conservatively. These trackers cannot allocate the target during significant deformations, but can re-detect it whenever the target assumes a stored appearance state. This is an important quality for long-term tracking, however, it cannot directly be applied to short-term tracking. Our approach draws on the long-term paradigm of conservative updates and uses it for conservative supervision of the part-based sub-model.

## 3. The proposed visual model

The proposed visual model is formalized as a hierarchically dependent set of the following sub-models: the holistic object templates (Section 3.1) representing the holistic detailed description, the global color model (Section 3.3) as holistic coarse representation and a deformable part-set (Section 3.2) as a local representation of the appearance. The proposed architecture is motivated by the observation that some segments of tracking session can be performed better using less free parameters, e.g. only position of the target, while other segments may require a more detailed description of the state – our visual model presents a mechanism to gracefully shift between these two modes.

**Localization.** A tracking iteration starts by initializing the tracker at a location predicted by a motion model estimated by a Kalman filter. The object templates are matched to the image, depending on the strength of the match of the best template, it can either provide a detection of the object (Figure 1, holistic), constrain the update process of the rest of the model (Figure 1, guided), or remain inactive (Figure 1, part-based). The part-based model is deformed to account for geometrical deformations and the color model generates the object segmentation mask. The resulting object location for the frame can be given either by the best template match (detection) or by the part-set model (otherwise).

**Update.** In the update step the part-based model is updated by removing and adding patches using a object segmentation mask generated by the color model and the estimated region of the object. In case of the consensus of the color and part-based models, a new template is considered for addition to the object template set. The color model is updated as well using the generated output region. The output region from the model is also used to update the motion model. Since it is clear that the holistic templates strongly influence the part-based and color-based representations, we refer to the our model model as an *anchored visual model* based on the fact that the templates are only used if they are deemed very reliable and in this case act as anchors to the rest of the model.

### 3.1. The object templates

The memory-system like representation contains a set of holistic templates of objects appearance acquired at different points in time $\mathcal{T}_t = \{T_1, T_2, \ldots\}$. In our implementation, the kernelized correlation filters with HOG features [12] are used as a representation, however, in contrast to [12], templates are not updated during tracking. At each frame, all templates are matched within a search region and a candidate with the maximal response is taken as the candidate detection, i.e.

$$\hat{T}_t = \arg \max_{T \in \mathcal{T}_{t-1}} d(T, \mathbf{Y}_t), \tag{1}$$

where $\hat{T}_t$ denotes the candidate template *anchor*, $d(\cdot, \cdot)$ the matching function that returns the best response and $\mathbf{Y}_t$ the current frame. The response of the anchor template is used to determine how the template will be used. If the template response exceeds $\lambda_D$, the tracker *detects* the object in a pose that is represented by the template. The region provided by the template is considered as the target output region and is used for updating the rest of the model. If the response exceeds threshold $\lambda_G$, where $\lambda_G < \lambda_D$, the tracker is not very confident about the detection and the part-set is used as the output region of the tracker instead. On the other hand, the template is still used to *guide* the remaining visual model in adding new parts by providing a region of interest. Finally, if the response is below $\lambda_G$, the template is not used at all and the part-set model estimates the output bounding box, which is also used for updating the rest of the visual model.

This design allows the tracker not to rely on the templates for shorter periods of time and therefore supports a very conservative updating mechanism to maintain a reliable template set. A new template is constructed from a region proposed by an agreement of the part-set region and the color segmentation. This template is added to the list of potential templates and is only promoted into the set of active templates after the overlap with the output region of the tracker exceeds a predefined threshold $\lambda_T$ for $\Omega_T$ frames. The only template that is directly introduced into the template set is the one provided by the initialization bounding box.

### 3.2. The part-based representation

The purpose of the part-based model is to account for geometrical deformations of the target. The constellation of parts is defined by $\mathbf{X}_t = \{\mathbf{x}_t^{(i)}\}_{i=1:N_t}$ and $\mathbf{H}_t = \{\mathbf{h}^{(i)}\}_{i=1:N_t}$, where $\mathbf{x}_t^{(i)}$ is the position of the $i$-th part and $\mathbf{h}^{(i)}$ is its visual model, a static grayscale histogram.

**Localization.** At time-step $t$, the localization of the target is performed by a three-step matching algorithm. Firstly, the displacement of each part $\mathbf{v}_t^{(i)}$ is estimated by the Lucas-Kanade optical flow. Based on forward-backward validation [27] we partition parts into subset for which the optical flow can be estimated reliably, $P_t$, and the the rest, $K_t$. The state $\mathbf{X}_t$ is estimated by maximizing the probability of part positions $\mathbf{X}_t$ conditioned on the measurements $\mathbf{Y}_t$ and estimation from the previous time-step $\hat{\mathbf{X}}_t$, i.e.,

$$p(\mathbf{X}_t|\mathbf{Y}_t, \hat{\mathbf{X}}_{t-1}, P_t, K_t) \propto \prod_{i \in K_t} p(\mathbf{Y}_t|\mathbf{x}_t^{(i)}) p(\mathbf{x}_t^{(i)}|\epsilon_t^{(i)}) \prod_{i \in P_t} \delta(\mathbf{x}_t^{(i)}|\mathbf{v}_t^{(i)}), \tag{2}$$

where $\epsilon_t^{(i)}$ are the parts in the neighborhood of $\mathbf{x}_t^{(i)}$, obtained in the same manner as in [25], and $\delta(\mathbf{x}_t^{(i)}|\mathbf{v}_t^{(i)})$ is a

Dirac-delta positioned at the flow displacement. This decomposition is obtained by assuming that a part is conditionally independent from other parts except immediate neighbors and that visual likelihood of a part is conditionally independent from the other parts. The part visual likelihood is defined as

$$p(\mathbf{Y}_t|\mathbf{x}_t^{(i)}) = e^{-\rho_i(\mathbf{x}_t^{(i)})/\sigma_c}, \quad (3)$$

where $\rho_i(\mathbf{x}_t^{(i)})$ is the Hellinger distance between the part reference histogram and the histogram extracted at $\mathbf{x}_t^{(i)}$ and $\sigma_c$ is a constant. The geometric constraint is defined as

$$p(\mathbf{x}_t^{(i)}|\epsilon_t^{(i)}) = e^{-||\mathbf{x}_t^{(i)}-\tilde{\mathbf{x}}_t^{(i)}||^2/\sigma_g}, \quad (4)$$

where $\tilde{\mathbf{x}}_t^{(i)}$ is the position predicted by the neighbors and $\sigma_g$ is a constant. This prediction is obtained by computing a similarity transform between the neighbors $\epsilon_t^{(i)}$ from $\hat{\mathbf{X}}_{t-1}$ and the current positions of the neighbors. Secondly, the optimization of (2) is initialized with a global displacement estimated using a generalized Hough voting algorithm. Thirdly, the probability (2) is maximized by the Iterated Conditional Modes (ICM) algorithm [3], which iterates over the parts and for each part computes a new position as the expected position under the conditional from (2). Note that the iterations are only required for parts in $K_t$. The optimization typically converges in less than ten iterations. This makes the MAP optimization of (2) gracefully shifting between flock-of-features approach and constellation-constrained optimization depending on the situation. After the matching is complete, the region of the object is estimated as the smallest axis-aligned rectangle containing all parts.

**Update.** To account for appearance changes, redundant parts are removed from the set and new ones are added. Redundant parts are recognized by looking for a region of high part density – parts outside the region are assumed to be drifting and are removed. The region is estimated by applying a mean-shift mode detection on a kernel density estimate [14] with a uniform kernel on the part locations. To account for size changes, the size of the kernel is based on the estimated object size.

New parts are added to the set by using the target segmentation mask (see Section 3.3) as well as image properties that maximize the chance of good optical flow estimation. To balance good object coverage and position quality the following score function is used for part sampling:

$$q(\mathbf{x}) = H(\mathbf{x}) + \alpha_U U(\mathbf{x}), \quad (5)$$

where $H(\mathbf{x})$ is the Harris corner score at position $\mathbf{x}$, $U(\mathbf{x})$ is a periodic function[1] that enforces uniform coverage of sampling points in homogeneous regions and $\alpha_U$ is a mixing constant. Only local maxima of $q(\mathbf{x})$ that are not already covered by existing parts and are inside the segmentation mask are kept and are ordered according to the color similarity likelihood and at most $N_U$ new parts are added at every time-step to ensure a gradual adaptation. The part-set is initialized at the beginning of sequence in a similar manner, but without a limit on a number of parts.

[1] We use a two-dimensional cosine signal that produces a grid-like pattern.

### 3.3. The color model

The visual model also maintains a global color model of the target and the immediate neighborhood described by a foreground and background RGB histograms, i.e., $F_t$ and $B_t$. The color model is used to bridge a gap between the detailed holistic and part-based representations by generating segmentation mask used for sampling parts as well as serving as a constraint for conservative updates of the holistic template set.

Given an estimate of the target bounding box, the segmentation mask is estimated as follows. The estimated region is expanded by $\alpha_S$ to account for the scale uncertainty. Foreground and background histograms are backprojected into the expanded region resulting in two backprojection maps, which are further smoothed by a Gaussian kernel to enforce spatial coherence, resulting in foreground and background probability maps, $p(\mathbf{x}|\mathrm{F}_t)$ and $p(\mathbf{x}|\mathrm{B}_t)$, respectively. The foreground posterior is calculated at each pixel using the Bayes rule

$$p(\mathcal{F}_t|\mathbf{x}) = \frac{p(\mathcal{F})p(\mathbf{x}|\mathrm{F}_t)}{p(\mathcal{F})p(\mathbf{x}|\mathrm{F}_t) + (1-p(\mathcal{F}))p(\mathbf{x}|\mathrm{B}_t)}, \quad (6)$$

where $p(\mathcal{F})$ denotes the object prior. A likelihood threshold is estimated such that the ratio between the number of pixels exceeding this threshold within the estimated region and the expanded region is greater than $\lambda_S$, as illustrated in Figure 2. If such a threshold cannot be set, the discrimination between foreground and background cannot be determined reliably and the segmentation mask is empty. The mask is further post-processed to remove outlier components.

## 4. Evaluation

In the following, the proposed tracker will be referred to as *anchored tracking* (ANT). The ANT was evaluated on the recent VOT benchmarks, VOT2013 [16] and VOT2014 [17], which provide a fully annotated dataset, evaluation protocol and an evaluation toolkit along with the results of a large number of state-of-the-art trackers. The large number of trackers tested makes these benchmarks arguably the largest short-term tracking benchmarks to date.

**Dataset and the evaluation protocol.** The datasets consist of 16 (VOT2013) and 25 (VOT2014) manually annotated sequences that contain various challenging visual tracking scenarios such as severe illumination changes, object deformations, abrupt motion changes scale variation, camera

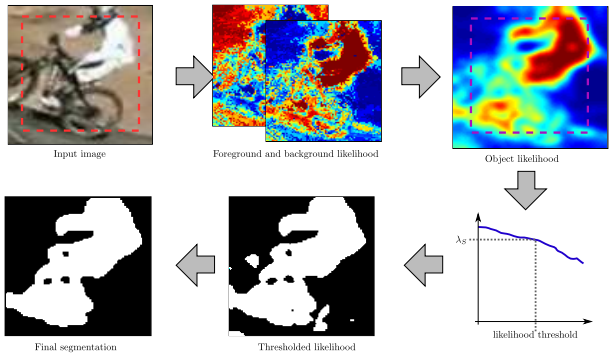


Figure 2. Segmentation mask $S_t(\mathbf{x})$ construction. Object likelihood at each pixel is estimated by histogram backprojection, followed by an automatic threshold selection using cumulative likelihood histograms and morphological post-processing.

motion and occlusion. The sequences were selected from a larger pool from various sources using a clustering methodology that provided a diverse dataset of reasonable size.

In the evaluation we follow the official protocol of VOT challenges, the trackers are initialized at the first frame using ground-truth annotations and reinitialized every time they drift away from the target. The results are summarized in terms of accuracy (average region overlap) and robustness (number of re-initializations). The experiments were performed using VOT toolkit which also provides ranking analysis that takes into account statistical and practical difference on accuracy and robustness performance measures to ensure a fair comparison. The details of the methodology are available in [16, 17, 15].

**Implementation and runtime performance.** The proposed tracker is implemented in a combination of Matlab and C++[2]. Despite the fact that some calculations are performed several times for implementation clarity the implementation performs at about five frames per second on a computer with AMD Opteron 6238 processor; an optimized native implementation would run in real-time on an average modern computer.

**Parameter configuration.** As required by the VOT evaluation protocol, the parameters of the tracker were fixed for all the experiments. The parts were modeled using 16-bin grayscale histograms, matching parameters were set to $\sigma_c = 3$ and $\sigma_g = 3$, update parameters to $\alpha_U = 10^{-4}$, $N_U = 2$. The color model used 32-bin RGB histograms for the color model. The segmentation parameters were set to $\lambda_S = 0.9$, $p(\mathcal{F}) = 0.4$. The template set was configured to use HOG-based correlation templates with $4 \times 4$ pixels and 9 orientations per cell and Gausssian kernel with $\sigma = 0.6$. The learning rate was controlled by $\Omega_T = 7$ and $\lambda_T = 0.8$. The operation mode thresholds were set to $\lambda_D = 0.85$ and $\lambda_G = 0.5$. We have experimentally analyzed model parameters and determined that they do not significantly impact tracking performance for small changes. The parameters that have the largest impact on model behavior are $\lambda_D$ and $\lambda_G$, which is expected as they directly influence the selection of the tracking mode. Lowering $\lambda_D$ decreases tracker robustness as it allows entering detection mode with less reliable template matching scores. Raising $\lambda_G$ decreases the effects of guide mode which lowers the accuracy of the tracker.

[2]The source code of the tracker as well as raw results of the evaluation are available at `http://go.vicos.si/ant`.

### 4.1. VOT2013 benchmark results

**Benefits of the proposed guiding mechanism.** Several variations of the ANT have been evaluated to test the contributions of each sub-model and the proposed cross-sub-model interaction: The ANT-D tracker is a tracker that only uses a single anchor template, ANT-P uses only parts and segmentation, and ANT-DP uses part-set together with the memory system, but the system only acts as a detection mechanism, i.e., it does not guide the update of the part set. The results are shown in Table 1.

Table 1. Average overlap and average number of failures on VOT2013 benchmark for ANT tracker derivations.

| | **ANT-D** | **ANT-P** | **ANT-DP** | **ANT** |
|---|---|---|---|---|
| **Overlap** ↑ | 0.64 | 0.39 | 0.46 | 0.64 |
| **Failures** ↓ | 2.39 | 0.88 | 0.39 | 0.00 |

As seen in Table 1, the ANT-D tracker achieves good accuracy, mainly at the expense of robustness since a single static template cannot properly address the appearance changes. On the other hand ANT-P achieves good robustness, but the accuracy is low since the part-based model applies self-supervised updating without external supervision and recovery capability from the template memory system. The ANT-DP combines the traits of ANT-D and ANT-P trackers, and benefits from switching between the detection and part-based tracking. The complete ANT tracker improves performance in terms of accuracy and robustness by using anchor templates not only to detect the object, but also to guide the update process of the part-set even if the template detection is not reliable enough for a full detection. In particular, ANT improves over the variations ANT-D, ANT-P and ANT-DP in accuracy as well as robustness. The results thus clearly support our hypothesis that the proposed combination of part-based visual model and holistic visual model improves the overall tracking performance.

**Comparison to the state-of-the-art.** The ANT was further compared to several top-performing state-of-the-art trackers [27, 25, 10, 20, 9] and baseline trackers [22, 1, 11, 34, 13, 30] on the VOT2013. The results of the evaluation are presented in Figure 3 as raw A-R plot [26], sequence-normalized and attribute-normalized ranking plots. The results are summarized in Table 2, per-sequence scores are available in the supplementary material.

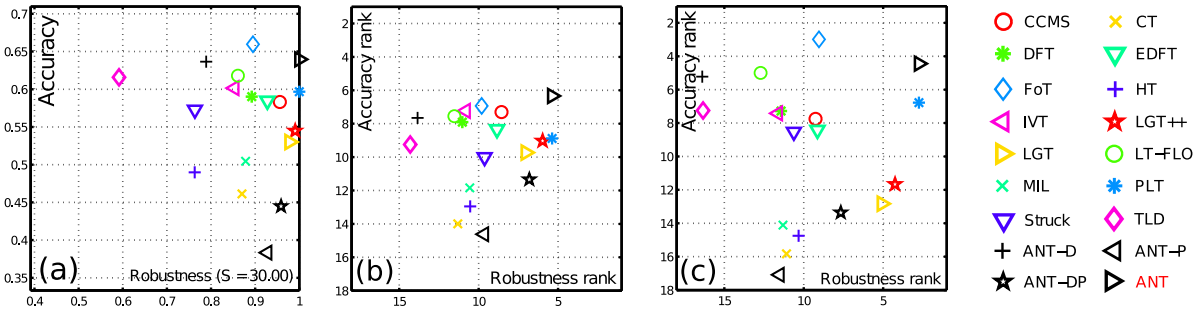


Figure 3. The VOT2013 A-R plots (a), sequence-normalized rank plots (b) and attribute-normalized rank plots (c). Trackers closer to the top-right corner are better performing.

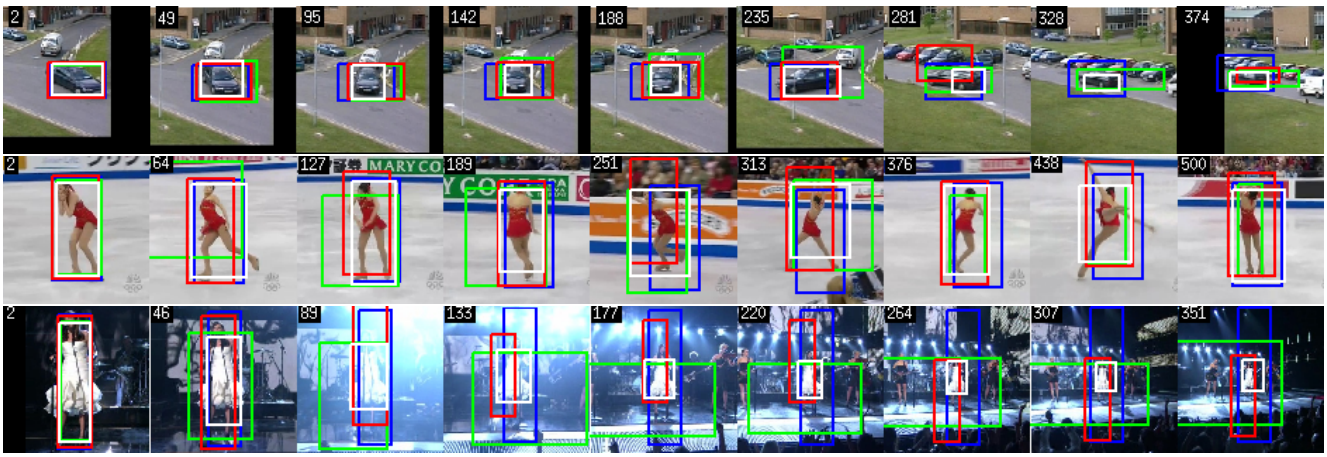

Figure 4. Visual comparison of trackers CCMS (red), LGT (green), PLT (blue), and ANT (white) on sequences *car iceskater*, *singer* from VOT2013 dataset.

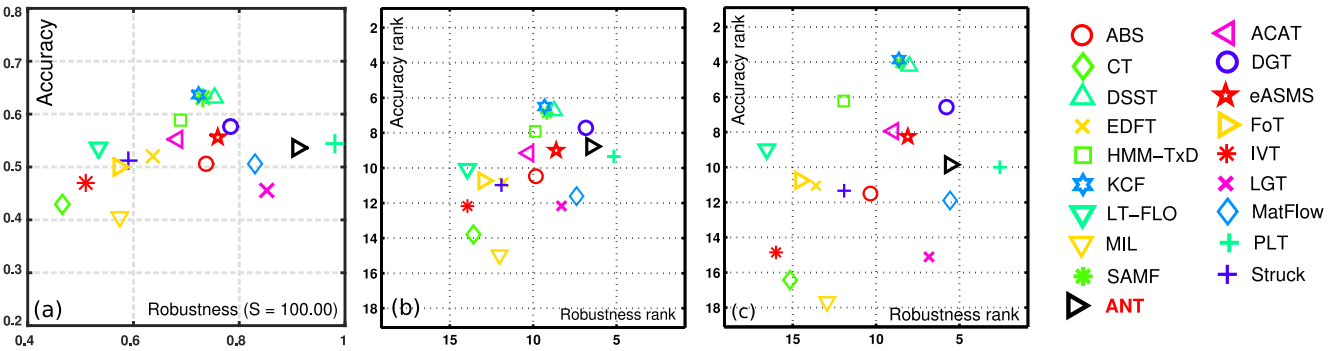


Figure 5. The VOT2014 A-R plots (a), sequence-normalized rank plots (b) and attribute-normalized rank plots (c). Trackers closer to the top-right corner are better performing.

Results in Figure 3 show that the proposed tracker outperforms all reference trackers by achieving the best accuracy and robustness. It is important to note that some trackers, like FoT [27] and LT-FLO [20], achieve a higher accuracy due to many re-initializations, which is not the case with ANT. The proposed tracker shares the first place with the winner of VOT2013 challenge, PLT, in robustness. Both trackers do not fail on any sequence, but PLT achieves a lower accuracy in case of deformations and scale changes due to its holistic visual model. As seen in Table 2, most of the other holistic trackers, like IVT, Struck, and EDFT are less robust in tracking non-rigid objects, but achieve a higher accuracy in comparison to the part-based LGT and LGT++, which are related to ANT. On the other hand, the ANT achieves an accuracy comparable to the holistic models and simultaneously outperforms related part-based trackers in robustness. This can be also observed on qualitative examples in Figure 4, e.g. in sequence *singer* where ANT better estimates the size of the object despite illumination and scale changes that are challenging for both part-

based LGT as well as holistic PLT and CCMS. This further confirms our hypothesis about retaining the best properties from holistic and part-based trackers.

### 4.2. VOT2014 benchmark results

The ANT was further evaluated on the most recent VOT2014 benchmark and compared to the state-of-the-art top performing [12, 7, 5, 25, 31, 28] and several baseline trackers [22, 1, 11, 34]. The results of the evaluation are presented in Figure 5 as raw A-R plot [26], sequence-normalized and attribute-normalized ranking plots, in Table 2 as well as in the supplementary material.

According to the results, ANT is the second most robust tracker. It is outperformed only by the PLT tracker. As we can see in Figure 6 most of this gain comes from the decreased robustness of ANT in case of occlusions. On the other hand the ANT tracker outperforms PLT in accuracy in case of size and illumination change as well as frames without degradation.

In terms of accuracy, the proposed tracker performs comparably to most reference trackers. The relative decrease in comparison to VOT2013 can be attributed to a more competitive set of trackers and to some degree to different annotation format, that uses rotated bounding boxes, which reduces region overlap with axis-aligned bounding boxes, reported by ANT. Trackers like DGT achieve better accuracy by utilizing computationally expensive segmentation. Holistic trackers, like DSST, KCF and SAMF perform better in accuracy, at a relative difference of about $10\%$, but fail approximately four times more often. This means that they are also more often reinitialized which consequently corrects the region estimate, resulting in increased final overlap.

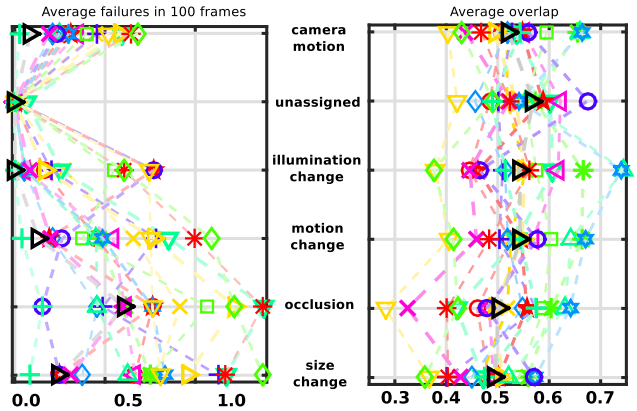


Figure 6. The attribute ordering plots for the VOT2014 benchmark. In both cases values that are closer to the outer border of the figure are better.

As seen in Figure 5, the ANT is ranked similarly in robustness as the part-based DGT. The raw results available in the supplementary material reveal that the DGT in fact fails approximately four times more often, but only on sequences with certain visual degradations. The DGT failures occur in sequences where the assumptions required for efficient color segmentation are violated. The hybrid nature of the proposed visual model in ANT is robust to a wide array of visual degradations as seen in Figure 6, hence the lower failure count. The ANT also outperforms the related LGT tracker in accuracy at relative increase of approximately $10\%$ and significantly outperforms it in robustness. This means that improved accuracy can in fact be attributed to increased robustness and not a trade-off between the two tracking aspects.

## 5. Conclusion

In this paper we have proposed a new model for visual tracking. The model is composed of several interacting types of visual sub-models that differ in the level of detail by which they describe the target. We propose to use holistic detailed, holistic coarse and part-based sub-models that mutually interact in localization and cross-sub-model updates by accounting for the potential uncertainty of the visual information. This makes the visual model shift between purely holistic and part-based behavior, depending on the visual uncertainty. A tracker that uses the proposed visual model was evaluated on challenging VOT2013 [16] and VOT2014 [17] benchmarks where we have shown that the mutual interaction between the sub-models significantly improves the performance of the tracker. We have compared the tracker to the state-of-the-art reference trackers, the results confirm the hypothesis that the proposed tracker outperforms related part-based trackers as well as many other trackers both in accuracy and robustness.

Furthermore, the template anchors concept in our tracker is very general and can easily be replaced by object-class-specific detectors like face detectors to aid tracking in specific applications like face tracking. Our ongoing work is directed towards exploring various possibilities of introducing application-specific priors into the tracker as well as detecting partial occlusions by observing changes in the part-based model.

**Acknowledgments:** This research was in part supported by ARRS projects L2-6765 and P2-0214.

## References

[1] B. Babenko, M.-H. Yang, and S. Belongie. Robust Object Tracking with Online Multiple Instance Learning. *IEEE TPAMI*, 33(8):1619–1632, Aug. 2011.

[2] V. Badrinarayanan, P. Perez, F. Le Clerc, and L. Oisel. Probabilistic Color and Adaptive Multi-Feature Tracking with Dynamically Switched Priority Between Cues. In *ICCV 2007*, pages 1–8, 2007.

Table 2. Results overview for VOT2013 and VOT2014 benchmarks in terms of average overlap and number of failures.

| VOT2013 | **HT** [10] | **CCMS** | **EDFT** [9] | **FoT** [27] | **IVT** [22] | **LGT++** [30] | **LGT** [25] | **PLT** | **Struck** [11] | **ANT** |
|---|---|---|---|---|---|---|---|---|---|---|
| **Overlap** ↑ | 0.49 | 0.61 | 0.60 | 0.64 | 0.61 | 0.56 | 0.54 | 0.61 | 0.53 | 0.64 |
| **Failures** ↓ | 4.25 | 0.56 | 0.79 | 1.54 | 1.62 | 0.08 | 0.26 | 0.00 | 3.58 | 0.00 |
| VOT2014 | **ACAT** | **SAMF** [31] | **MatFlow** | **eASMS** [28] | **DSST** [7] | **KCF** [12] | **DGT** [5] | **PLT** | **LGT** [25] | **ANT** |
| **Overlap** ↑ | 0.55 | 0.64 | 0.51 | 0.56 | 0.63 | 0.64 | 0.58 | 0.54 | 0.46 | 0.54 |
| **Failures** ↓ | 1.35 | 0.92 | 0.57 | 1.32 | 0.84 | 0.99 | 1.15 | 0.05 | 0.62 | 0.27 |


[3] J. Besag. On the Statistical Analysis of Dirty Pictures. *J. R. Stat. Soc. Ser. B*, 48(3):pp. 259–302, 1986.

[4] W. Bouachir and G.-A. Bilodeau. Structure-aware keypoint tracking for partial occlusion handling. In *WACV2014*, pages 877–884. IEEE, 2014.

[5] Z. Cai, L. Wen, Z. Lei, N. Vasconcelos, and S. Z. Li. Robust Deformable and Occluded Object Tracking With Dynamic Graph. *IEEE Trans. Image Process.*, 23(12):5497–5509, 2014.

[6] W.-Y. Chang, C.-S. Chen, and Y.-P. Hung. Tracking by Parts: A Bayesian Approach With Component Collaboration. *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, 39(2):375–388, 2009.

[7] M. Danelljan, F. S. Khan, M. Felsberg, J. V. de Weijer, G. Häger, F. S. Khan, and M. Felsberg. Accurate scale estimation for robust visual tracking. In *BMVC 2014*, 2014.

[8] S. Duffner and C. Garcia. PixelTrack: A Fast Adaptive Algorithm for Tracking Non-rigid Objects. In *ICCV*, Dec. 2013.

[9] M. Felsberg. Enhanced Distribution Field Tracking using Channel Representations. In *ICCV Work.*, 2013.

[10] M. Godec, P. M. Roth, and H. Bischof. Hough-based Tracking of Non-rigid Objects. *CVIU*, 117(10):1245–1256, 2013.

[11] S. Hare, A. Saffari, and P. H. S. Torr. Struck: Structured output tracking with kernels. In *ICCV 2011*, pages 263–270. IEEE, Nov. 2011.

[12] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. High-Speed Tracking with Kernelized Correlation Filters. *IEEE TPAMI*, 2014.

[13] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE TPAMI*, 34(7):1409–1422, 2012.

[14] M. Kristan, A. Leonardis, and D. Skočaj. Multivariate online kernel density estimation with gaussian kernels. *Pattern Recognition*, 44:2630–2642, 2011.

[15] M. Kristan, J. Matas, A. Leonardis, T. Vojir, R. Pflugfelder, G. Fernandez, G. Nebehay, F. Porikli, and L. Cehovin. A novel performance evaluation methodology for single-target trackers. *arXiv preprint arXiv:1503.01313*, 2015.

[16] M. Kristan, R. Pflugfelder, A. Leonardis, J. Matas, F. Porikli, L. Čehovin, G. Nebehay, G. Fernandez, T. Vojir, and et al. The Visual Object Tracking VOT2013 challenge results. In *ICCV Work.*, pages 98–111, 2013.

[17] M. Kristan, R. Pflugfelder, A. Leonardis, J. Matas, L. Čehovin, G. Nebehay, T. Vojir, G. Fernández, and et al. The Visual Object Tracking VOT2014 challenge results. In *ECCV Work.*, 2014.

[18] J. Kwon and K. M. Lee. Tracking by Sampling Trackers. In *ICCV*, pages 1195–1202. IEEE, Nov. 2011.

[19] J. S. Kwon and K. M. Lee. Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive Basin Hopping Monte Carlo sampling. In *CVPR 2009*, pages 1208–1215, 2009.

[20] K. Lebeda, S. Hadfield, J. Matas, and R. Bowden. Long-Term Tracking Through Failure Cases. In *ICCV Work.*, 2013.

[21] F. Pernici and A. D. Bimbo. Object Tracking by Oversampling Local Features. *IEEE TPAMI*, 2013.

[22] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang. Incremental Learning for Robust Visual Tracking. *IJCV*, 77(1-3):125–141, May 2008.

[23] J. Santner, C. Leistner, A. Saffari, T. Pock, and H. Bischof. PROST - Parallel Robust Online Simple Tracking. In *CVPR 2010*, San Francisco, CA, USA, 2010.

[24] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah. Visual Tracking: an Experimental Survey. *TPAMI*, 2013.

[25] L. Čehovin, M. Kristan, and A. Leonardis. Robust Visual Tracking using an Adaptive Coupled-layer Visual Model. *TPAMI*, 35(4):941–953, Apr. 2013.

[26] L. Čehovin, M. Kristan, and A. Leonardis. Is my new tracker really better than yours? In *WACV 2014*, pages 540–547. IEEE, 2014.

[27] T. Vojír and J. Matas. Robustifying the Flock of Trackers. In A. Wendel, S. Sternig, and M. Godec, editors, *CVWW 2011*, pages 91–97, Inffeldgasse 16/II, Graz, Austria, 2011. Graz University of Technology.

[28] T. Vojir, J. Noskova, and J. Matas. Robust scale-adaptive mean-shift for tracking. *Pattern Recognit. Lett.*, 49:250–258, 2014.

[29] Y. Wu, J. Lim, and M.-h. Yang. Online Object Tracking: A Benchmark. In *CVPR*, 2013.

[30] J. Xiao, R. Stolkin, and A. Leonardis. An Enhanced Adaptive Coupled-Layer LGTracker++. In *ICCVW 2013*, pages 137–144, Dec. 2013.

[31] X. Yang, Q. Xiao, S. Wang, and P. Liu. Real-time Tracking via Deformable Structure Regression Learning. *ICPR 2014*, 2014.

[32] Z. Yin and R. Collins. On-the-fly Object Modeling while Tracking. In *CVPR 2007*, pages 1–8, 2007.

[33] H. Zhang, S. Cai, and L. Quan. Real-Time Object Tracking with Generalized Part-Based Appearance Model and Structure-Constrained Motion Model. *ICPR 2014*, 2014.

[34] K. Zhang, L. Zhang, and M.-H. Yang. Real-time Compressive Tracking. In *ECCV 2012*, 2012.

[35] H. Zhou, Y. Yuan, and C. Shi. Object tracking using SIFT features and mean-shift. *CVIU*, 113(3):345–352, 2009.