Supporting Annotation of Anatomical Landmarks using Automatic Scale Selection

Sebastian Bernd Krah, Jürgen Brauer, Wolfgang Hübner, Michael Arens

Fraunhofer Institute of Optronics, System Technologies and Image Exploitation
{sebastian.krah, juergen.brauer, wolfgang.huebner, michael.arens}@iosb.fraunhofer.de

Abstract. The effectiveness of appearance based person models strongly relies on a sufficiently large number of high quality training samples. Generating training data in terms of bounding boxes is already a time consuming task. If more complex person models are used, like part-based models or models suitable for human pose estimation, the labeling process becomes infeasible. In the context of pose estimation, motion capturing is often used to generate ground truth data. A major problem with this approach is that motion capturing is usually done in artificial environments with only few persons. It is therefore difficult to generate classifiers which are able to localize anatomical landmarks on a moving person. In order to solve this problem we propose a solution to generate annotations of anatomical landmarks using a semi-automatic work flow, based on tracking and automatic scale selection.
The contribution of the paper is twofold. First, different tracking methods are evaluated in terms of their properties to follow anatomical structures on a moving person. Second, in order to determine the spatial extents of anatomical landmarks some simple but effective scale selection methods are proposed. The resulting person models are intended to generate a suitable basis for learning regression models for monocular pose estimation, as well as for training part-based models directly. Results of a comprehensive quantitative evaluation on the UMPM dataset are presented, while we also show examples of qualitative results on two challenging YouTube sequences.

Keywords: Semi-automatic annotation, Tracking of anatomical landmarks, Automatic scale selection

1 Introduction

For training part-based person detectors or 2D human pose estimators, learning-based approaches often need training examples of the form (person image, 2D ground-truth landmark regions) (e.g., [4], [10], [5]). Unfortunately, such training data is currently only available for some few videos which were recorded with marker-based motion capture systems (e.g., UMPM [1], HumanEva [9]). For this, it is desirable to be able to annotate selected training videos that are similar to the image material during the envisaged application manually. But labeling a large amount of video frames only by means of manual annotation is time-consuming and laborious since the ground truth landmark regions have to be annotated in every video frame, e.g., by drawing a rectangle around each of N landmarks. Here we want to generate such annotations automatically

This is the author's version of the paper. Published at AMDO 2014 http://amdo2014.uib.es/
VIII Conference on Articulated Motion and Deformable Objects. Palma, Mallorca, Spain. 16–18 July 2014
The final publication is available at link.springer.com in the conference proceedings:
F.J. Perales and J. Santos-Victor (Eds.): AMDO 2014, LNCS 8563, pp. 61–70, 2014.

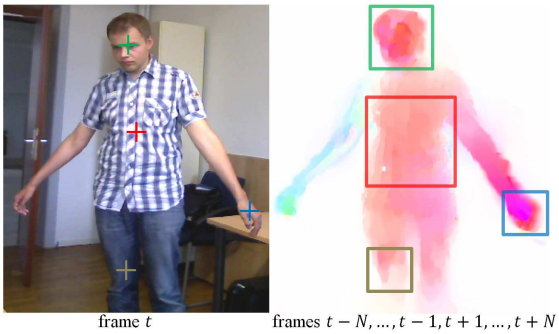


Fig. 1: **Semi-automatic landmark annotation.** Left: manual annotation of 2D locations of anatomical landmarks for an anchor frame t. Right: using the manual annotation information we speed-up the ground-truth labeling process by first estimating the spatial extent of each landmark automatically and then track the landmarks to neighbored frames using variants of optical flow (visualized here: TV-L1 optical flow) and appearance based online tracking methods adapted to the task of landmark tracking.

starting from so-called *weakly annotated anchor frames*, which are frames annotated by the user with the limitation that the user only labels the center location of each landmark, e.g., by a single point click. Determining the 2D region of each landmark and tracking the landmarks backwards and forwards in the video for some video frames will be done automatically and is the topic of this paper. There are two main contributions by this paper. First, we present a comparative evaluation of an appearance based visual tracker and different optical flow based tracking approaches in order to better assess which of both approaches is more appropriate for the task of landmark tracking. Second, we propose three methods that allow to estimate the scale of each landmark automatically, thereby lifting up the weak landmark annotation of the user by a 2D point to a full 2D rectangular region annotation. Together, this allows to annotate ground-truth data in a semi-automatic fashion, more than 18 times faster compared to a manual annotation, if automatic generated landmarks that do not deviate more than 3% of the object height from the true landmark locations are considered to be acceptable as training data.

Related Work. Tracking is a very active research area with many approaches presented in the last decades. [14] and [13] provide surveys of tracking approaches. Among the different appearance based tracking approaches, online tracking approaches, which update an appearance model of the object on-the-fly, have shown to be successful even in cases of strong appearance changes of the object. [12] provides an exhaustive evaluation of 25 current state-of-the-art online trackers, which renders the Compressive Tracking (CT) approach by Zhang et al. [16] as currently the best online tracking method. We therefore choose CT here as a representative for the class of appearance based trackers. Among optical flow based object tracking methods, dense (e.g.,[8]) and sparse optical flow (e.g., [7]) has been used. While dense optical flow methods are computational more demanding compared to sparse flow methods, they provide a higher accuracy. Since we do not focus on real-time ground-truth generation of training data, but allow for off-line annotation, we select the TV-L1 method [15] as a representative and basis for optical flow based tracking. Ground truth training data is needed by many computer vision al-

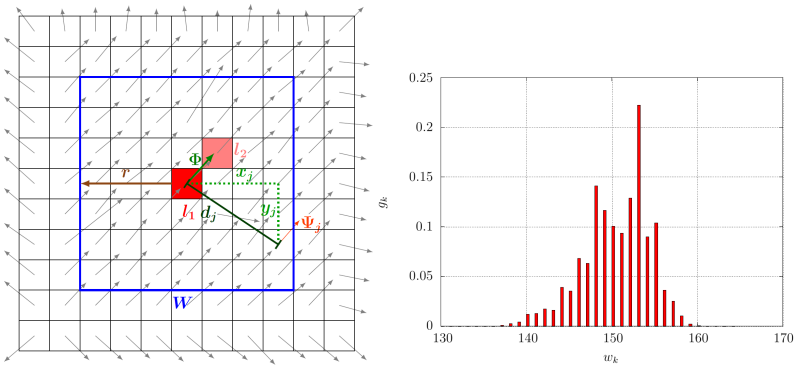


Fig. 2: Left: optical flow vector field. For the non-histogram based methods, the location of the landmark is predicted for the next frame using all optical flow vectors within a region W. Right: exemplary cutout of an angle-weight histogram.

gorithms. Offline (e.g., ViPER [3]) and online labeling tools (e.g., LabelMe [6]) have been published, allowing to annotate, e.g., bounding boxes or polygons. Nevertheless, semi-automatic annotation is often restricted to linear interpolation which assumes constant velocity of the object. The VATIC annotation tool [11] is one of few tools that use object tracking in order to ease the annotation process. The tool uses a feature descriptor composed of a HOG descriptor and color features to model the appearance of manually annotated objects. For tracking a linear SVM is used. Nevertheless, a fixed appearance model of the object with a fixed spatial extent is not appropriate to track objects in cases of strong appearance changes, e.g., if the object size changes or in the case of in-depth-rotations. [2] addresses the task of dealing with partially labeled data as well. There the task is to learn a part-based model using images labeled only with bounding boxes around the objects. For this, part locations which are not labeled are treated as latent variables during the training procedure. In contrast to [2] here we consider the situation that parts are labeled, but only the 2D locations, while the 2D landmark regions are unknown.

In section 2 we present the details about the optical flow and appearance based tracking methods used to propagate the user annotation from the anchor frame backwards and forwards within a video. Section 3 introduces three methods to determine the size of a landmark automatically. The results of the comparative evaluation of optical flow and appearance based landmark tracking and the different landmark scale estimation methods are presented in section 4, while section 5 presents the conclusions.

2 Tracking of Landmarks

For tracking a landmark from a frame t_1 – where we already have an estimate of the location $\boldsymbol{l_1}$ of the landmark – to a frame t_2 with unknown location $\boldsymbol{l_2}$, we compute a prediction vector $\boldsymbol{\Phi}$, that describes the translation of the landmark, i.e., $\boldsymbol{l_2} = \boldsymbol{l_1} + \boldsymbol{\Phi}$.

Optical Flow Based Tracking of Landmarks (OFT). For two consecutive video frames t_1 and t_2 we compute the TV-L1 optical flow, i.e., an optical flow field which contains information about the movement of every pixel (see Fig. 2 left). The following

methods predict the landmark location using TV-L1 optical flow information and can be divided into non-histogram and histogram based methods.

OFT with Non-Histogram Based Methods. The first two methods considered compute $\boldsymbol{\Phi}$ based on a weighted average of the optical flow vectors within a region W:

$$\boldsymbol{\Phi} = \left(\sum_{\boldsymbol{d_j} \in W} g(\boldsymbol{d_j}) \cdot \boldsymbol{\Psi_j}\right) \cdot \left(\sum_{\boldsymbol{d_j} \in W} g(\boldsymbol{d_j})\right)^{-1}$$

Here $\boldsymbol{\Psi_j} = (u_j, v_j)^\mathsf{T}$ represents the optical flow vector at location $\boldsymbol{d_j} = (x_j, y_j)^\mathsf{T}$ relative to $\boldsymbol{l_1}$. A weighting function g is used to weight each of the optical flow vectors in dependence on its location in W relative to l_1. With $g(\boldsymbol{d_j}) = 1$ we give each of the optical flow vectors in W the same weight, i.e., $\boldsymbol{\Phi}$ is the arithmetic mean of all optical flow vectors in W (method name: Φ). Since the spatial extent of a landmark is unknown it is more promising to weight optical flow vectors which are close to the considered landmark larger than those which are far away from l_1. This can be achieved by using a Gaussian kernel weighting function $g(\boldsymbol{d_j}) = \mathcal{N}\left(\|\boldsymbol{d_j}\|_2, r/3\right)$ (method name: Φ_G).

OFT with Histogram Based Methods. The non-histogram based methods do not take into account that (i) some of the optical flow vectors might be erroneous due to outliers or (ii) optical flow vectors can point into different directions if the region W contains image structures of other landmarks that move into other directions. The following histogram based approaches offer the possibility to tackle the problem of outliers (i) and contradictory optical flow information (ii). In the following, optical flow vectors are described by their length η^j and their angle ω^j, i.e., $\boldsymbol{\Psi_j} = \left(\eta^j, \omega^j\right)^\mathsf{T}$ instead of their x- and y-translation components. Both following two methods first compute an angle-weight histogram that represents the information how often each angle ω^j occurs when considering all optical flow vectors in W, weighted by the lengths of the corresponding optical flow vectors. The angle-weight histogram is discretized into bins $\omega_k, (k = 1, ..., M)$ and for each angle bin a weight g_k is maintained. Each optical flow vector $\boldsymbol{\Psi_j}$ casts a vote into the next bin ω_k that corresponds to its angle ω^j, where the vote strength is set to η^j. Hence $g_k = \sum_{j=0, \omega_k = \omega^j}^{N-1} g(\boldsymbol{d_j}) \cdot \eta^j$. Here $N \in \mathbb{N}$ represents the number of optical flow vectors in W. In Fig. 2 (right) we present an example of the computed weights g_k for some of the angle bins ω_k of such a angle-weight histogram computed for a region W of a real frame. The angle-weight histogram shows a clear peak which corresponds to a favored direction of the optical flow vectors within W. The idea to compute $\boldsymbol{\Phi} = (\omega_\phi, l_\phi)^\mathsf{T}$ is to use only the optical flow vectors $\boldsymbol{\Psi_j}$ that belong to this favored direction. The favored direction corresponds to the bin ω_ϕ where the peak can be localized and the length of the landmark prediction vector is computed by:

$$l_\phi = \left(\sum_{j=0, \omega_\phi = \omega^j}^{N-1} g(\boldsymbol{d_j}) \cdot \eta^j\right) \cdot \left(\sum_{j=0, \omega_\phi = \omega^j}^{N-1} g(\boldsymbol{d_j})\right)^{-1}$$

Here the length of each optical flow vector is weighted by a weighting function g again. We consider two variants in the following: (i) a weighted mean of the lengths η^j of all optical flow vectors that voted into bin ω_ϕ, i.e., $g(\boldsymbol{d_j}) = 1$ (method name: Φ^H) and (ii) weighting each optical flow vector length by its distance to the landmark, i.e., $g(\boldsymbol{d_j}) = \mathcal{N}\left(\|\boldsymbol{d_j}\|_2, r/3\right)$ (method name: Φ^H_G).

Appearance Based Tracking of Landmarks (CT). As described in section 1 we use the Compressive Tracker (CT) [16] as a representative for the appearance based methods. CT is an online tracker, i.e., starting from a selected image region, it generates

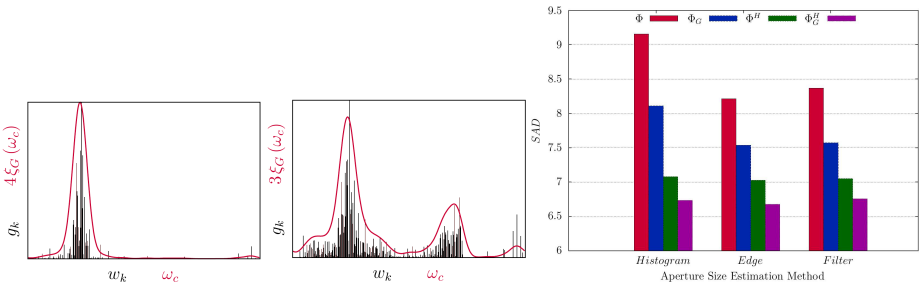


Fig. 3: Left+Mid: Angle-weight histograms for a landmark using a too small (left: one peak) and a too large landmark region (mid: two peaks). g_k: discrete histogram, ξ_G: continuous histogram using a Gaussian kernel density estimator. Right: Tracking errors for different combinations of optical flow and scale estimation methods

an appearance model (AM) and updates the AM on-the-fly while tracking. In contrast to the optical flow based methods, we cannot recompute a new landmark region radius r for each new frame. Instead we compute the region size only once using the scale estimation methods described in the next section when we initialize a CT online tracker for each landmark in the anchor frames. The reason is that the AM that is established and updated for each landmark is not scale-invariant, i.e., the AM can only describe a fixed region size. For CT we use the reference implementation provided by the authors[1]. In the following we use ζ to denote CT related tracking results.

3 Automatic Scale Estimation

Due to the weak labeling scenario, the actual landmark region is not specified by the user, but only its 2D center location. In the following three methods are presented that can be used to estimate the spatial extent of a landmark automatically. The corresponding landmark region can then be used by the optical flow methods to estimate a landmark prediction vector or by the CT to update its AM.

Histogram Based (H). The motivation behind the first approach is that a too small or an appropriate landmark region radius will result in a angle-weight histogram with a single peak (Fig. 3 left), since only optical flow vectors of the corresponding landmark are included, i.e., image regions that consistently move into a single direction. If the region radius is too large, image structures of other landmarks will be included, i.e., some optical flow vectors in that region will point into a second direction – as long as these other image structures do not move into the same direction – and a second peak will emerge in the histogram (Fig. 3 middle). A rough estimate for the landmark can be computed therefore by starting with a small region radius r_0 and increase it incrementally by Δr until a second large peak occurs in the angle-weight histogram at radius r_1 and take $r = r_1 - \Delta r$ as a region radius estimate. Since the detection of local maxima in a discrete angle-weight histogram turns out to be not reliable enough, we use a Gaussian kernel density estimator $\kappa_G(\omega_k)$ to compute a continuous density estimate on basis of the discrete histogram.

[1] `http://www4.comp.polyu.edu.hk/~cslzhang/CT/CT.htm`

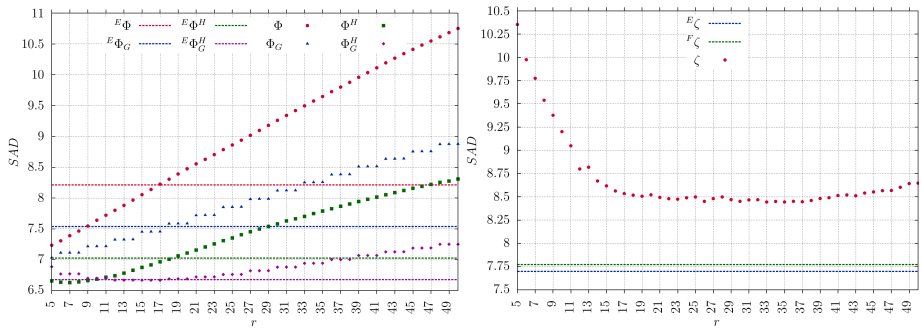


Fig. 4: **Tracking errors for constant vs. estimated landmark scale.** The plots show the landmark tracking errors for the different optical flow (left image, Φ) and CT (right image, ζ) based landmark tracking methods using constant scale (of 5-49 pixels) or dynamically estimated landmark scales based on the edge-based (E) or the filter-based (F) aperture size estimation. For easier comparison with the constant scale results the tracking errors using the dynamic scale estimation are shown as lines.

Edge Based (E). The central idea of this method is that based on an edge image the region radius should be selected such that at least a minimum amount of edge pixels are contained in the region. Assuming the landmark region is more or less homogeneously, edge pixels will occur at the borders of the region. More precisely, we compute a Canny edge image, start with a region radius r_0 and increase it incrementally by Δr. For each region radius the number of edge pixels $E(r)$ in the corresponding region W is compared with a threshold Θ. The landmark region radius estimate is the first r, such that $E\left(r\right) > \Theta$.

Filter Based (F). The basic idea behind this third method is that the scale should be selected in such a way that the brightness of the region border differs significantly from its region center, similar to SIFT and SURF blob keypoint detection. Such a region radius can be computed by filtering image patches at the hypothesized landmark location $\boldsymbol{l_1} = (x, y)^\mathsf{T}$ with different blob filters, described each by a blob filter matrix:

$$B_r^{\boldsymbol{l_1}} = \begin{cases} {}^{+1}/_{N_+} & , \sqrt{(x - {}^{r}/_{2})^2 + (y - {}^{r}/_{2})^2} < {}^{r}/_{2} \\ {}^{-1}/_{N_-} & , \text{else} \end{cases}$$

N_+ and N_- are normalization factors such that the positive and negative elements of the blob filter matrix sum up to 1 each. We convolve the image patch at the hypothesized landmark location with blob filter matrices of different radii r and take as estimate for the landmark region the radius r for which we get the strongest filter response.

4 Evaluation

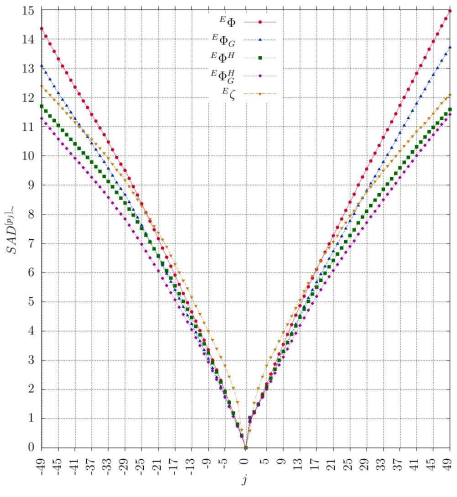
Quantitative Analysis. For the quantitative evaluation we choose the UMPM benchmark [1] since it provides 3D motion capture data together with camera calibration data. This allows to project the 3D landmark coordinates into the image, thereby generating ground truth 2D landmark center locations which can be compared with the automatically generated landmark locations by the different tracking approaches proposed here.

The manual annotation of *weakly annotated anchor frames* is simulated by using the provided ground truth information of landmarks every 99 frames of an UMPM video. Starting from an anchor frame the landmarks are tracked for 49 consecutive frames forwards and backwards. The tracking error is the average sum of absolute differences (SAD) between the UMPM ground truth landmark locations and the automatically generated landmark locations, where we average over all 15 landmarks considered and all evaluation frames. The distance between a ground truth and a tracked landmark location is measured in relative person (bounding box) height units and explicitly not in pixels in order to make the error measure independent of the displayed size of a person. Overall we used the 19 single person videos of the UMPM dataset, which corresponds to approx. 50 000 evaluation frames for each tracking method.

I. Dynamically estimated vs. constant scale. Fig. 4 shows the tracking error when we use a constant scale (of 5 to 49 pixels) for each of the 49 frames left and right to the anchor frame or a dynamically estimated scale, using the edge-based (E) or the filter-based (F) method. The results allow to draw two main conclusions. First, the optical flow histogram-based methods yield better tracking results than the simple averaging methods (compare, e.g., Φ, Φ_G with Φ^H, Φ_G^H). Second, the methods that estimate the landmark scale dynamically (with preceding E and F) yield better average tracking errors than the constant scale methods (without preceding E and F), which is most clearly shown for the case of the CT (bottom plot).

II. Comparison of scale estimation methods (H vs. E vs. F). Fig. 3 right compares the average tracking errors for the different optical flow methods (Φ, Φ_G, Φ^H, Φ_G^H) combined with the three different scale estimation methods (H,E,F), where each combination is evaluated on approx. 50 000 frames and 15 landmark locations estimated for each frame. The plot allows to draw two further conclusions. First, we can see a clear ranking of the four different optical flow methods w.r.t. the tracking error, namely: $\Phi > \Phi_G > \Phi^H > \Phi_G^H$. Second, there are no large significant differences in the errors depending on the scale estimation method, i.e., H $\approx$ E $\approx$ F.

III. Tracking error as a function of the distance to the anchor frame. In the figure at the right we show the average tracking error as a function of the distance to the anchor frame for the different optical flow based methods (Φ) and the CT ($^E\zeta$) using an edge-based scale estimation (E). The plot allows to answer the question how far we can track the landmarks to the left and right starting from an anchor frame (frame 0) if we allow for an average tracking error of maximally Θ percent. When accepting an average landmark tracking error of $\Theta = 5\%$ of the person height, we can use the annotated frames up to 17 frames left and right from the anchor frame without the need of any



further manual post-processing, i.e., for each weakly annotated anchor frame, we can generate 34 automatically annotated frames with estimated landmark centers and landmark regions.

IV. Optical flow vs. appearance based tracking. The best optical flow based landmark tracking method (${}^{E}\Phi_{G}^{H}$) that exploits the edge-based approach to estimate the landmark scale yields a tracking error of 6.7% and the CT based tracker with edge-based scale estimation an error of 7.7% (of the person height) when tracking 49 frames to the left and right, i.e., automatically annotating 98 frames given one weakly annotated anchor frame. This seems to indicate that there is no large difference between the optical flow and appearance based landmark tracking approaches proposed here.

Qualitative Analysis. In Fig. 5 we show some qualitative examples of tracking results on a UMPM sequence and two challenging YouTube sport sequences showing fast movements with motion blur and some background clutter in the case of the basketball sequence. For some anchor frames we manually labeled four different landmarks (head, shoulder, hand, foot) and used the optical flow (Φ_{G}^{H}) and the CT (ζ) tracking methods to track the landmarks. The landmark region scales were estimated using the E method and are depicted by the rectangles, while the black dot denotes the estimated landmark center. Note the large differences in the estimated landmark scales when considering different landmarks. The estimated scales of the head and the left shoulder are very similar for different frames and correspond to the extents of the head and the shoulder, which is interesting, since we do not have specified which image structures belong to the head or the shoulder anywhere. Remember that the user only labels the landmark centers and does not provide segmentation information for the landmarks. The scale corresponds to the landmark extents here since edge pixels occur at the borders to other landmarks (for the head at the border to the torso, and for the shoulder at the border to the head and the end of the T-shirt sleeves). The estimated scale for the foot often ends at the edge of the sock to the lower leg. For the badminton sequence which shows a relative homogeneous background the hand region ends typically at the elbow, while for the basketball sequence the hand region is significantly smaller, since there is much background clutter present that belongs to other image structures than the hand.

5 Conclusions

The paper addressed the task of supporting the generation of ground-truth landmark annotations by tracking anatomical landmarks on highly articulated objects given a few manually annotated anchor frames. We explored four different optical flow based methods and a state-of-the-art appearance based method (CT) in combination with three different simple scale selection methods, which are used to obtain a region from which to use the optical flow vectors to compute a landmark prediction vector or to update the appearance model of the CT tracker. Each method was evaluated on approx. 50 000 frames of the UMPM benchmark and the quantitative results show that there is no large difference between the simple optical flow based methods and the CT tracking method. Since a simple histogram-based detection of the main flow direction with Gaussian weighting (${}^{E}\Phi_{G}^{H}$) results in even slightly smaller tracking errors compared to the much more complex CT method, we propose to use this method for landmark tracking. Re-estimating

the scale of each landmark for each new frame yields significantly better tracking results than using a fixed scale, while no large differences between the three scale estimation methods (H,E,F) concerning the tracking errors were observed. Even though the optical flow based approaches in combination with one of the three scale estimation methods renders as a simplistic approach for landmark tracking, we can automatically annotate approx. 34 frames for each weakly annotated frame if average landmark localizations errors of up to 5% of the person height are acceptable and thereby speed-up the manual annotation process by a corresponding factor of 34 as well.

References

1. Aa, N.v.d., Luo, X., Giezeman, G., Tan, R., Veltkamp, R.: Utrecht Multi-Person Motion (UMPM) benchmark: a multi-person dataset with synchronized video and motion capture data for evaluation of articulated human motion and interaction. In: Proc. of Human Interaction in Computer Vision (HICV) workshop (2011)
2. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object Detection with Discriminatively Trained Part Based Models. IEEE Trans. on PAMI 32(9), 1627–1645 (2010)
3. Mihalcik, D., Doermann, D.: The Design and Implementation of ViPER. Tech. rep., University of Maryland (2003)
4. Mori, G., Malik, J.: Recovering 3D Human Body Configurations Using Shape Contexts. IEEE Trans. on PAMI 28(7), 1052–1062 (2006)
5. Müller, J., Arens, M.: Human Pose Estimation with Implicit Shape Models. In: Proc. of ACM ARTEMIS 2010. pp. 9–14. ARTEMIS '10, ACM, New York, NY, USA (2010)
6. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: LabelMe: A Database and Web-Based Tool for Image Annotation. Int. J. Comput. Vision 77(1-3), 157–173 (May 2008)
7. Salmane, H., Ruichek, Y., Khoudour, L.: Object Tracking Using Harris Corner Points Based Optical Flow Propagation and Kalman Filter. In: Proc. of 14th IEEE Intelligent Transportation Systems Conference (ITSC'2011). pp. 67–73. Washington D.C., USA (2011)
8. Schikora, M., Koch, W., Cremers, D.: Multi-Object Tracking via High Accuracy Optical Flow and Finite Set Statistics. In: Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP) (2011)
9. Sigal, L., Balan, A., Black, M.: HumanEva: Synchronized Video and Motion Capture Dataset and Baseline Algorithm for Evaluation of Articulated Human Motion. Int. Journal of Computer Vision 87(1), 4–27 (Mar 2010)
10. Sigal, L., Black, M.J.: Predicting 3D People from 2D Pictures. In: Proc. of Int. Conf. on Articulated Motion and Deformable Objects (AMDO). pp. 185–195 (2006)
11. Vondrick, C., Patterson, D., Ramanan, D.: Efficiently Scaling up Crowdsourced Video Annotation. Int. Journal of Computer Vision pp. 1–21 (2012), 10.1007/s11263-012-0564-1
12. Wu, Y., Lim, J., Yang, M.H.: Online Object Tracking: A Benchmark. In: Proc. of CVPR 2013 (2013)
13. Yang, H., Shao, L., Zheng, F., Wang, L., Song, Z.: Recent advances and trends in visual tracking: A review. Neurocomputing 74(18), 3823–3831 (Nov 2011)
14. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. ACM Computing Surveys 38(4) (2006)
15. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime TV-L1 optical flow. In: Pattern Recognition, pp. 214–223. Springer (2007)
16. Zhang, K., Zhang, L., Yang, M.H.: Real-time compressive tracking. In: Proc. of ECCV 2012. pp. 864–877. ECCV'12, Springer-Verlag, Berlin, Heidelberg (2012)

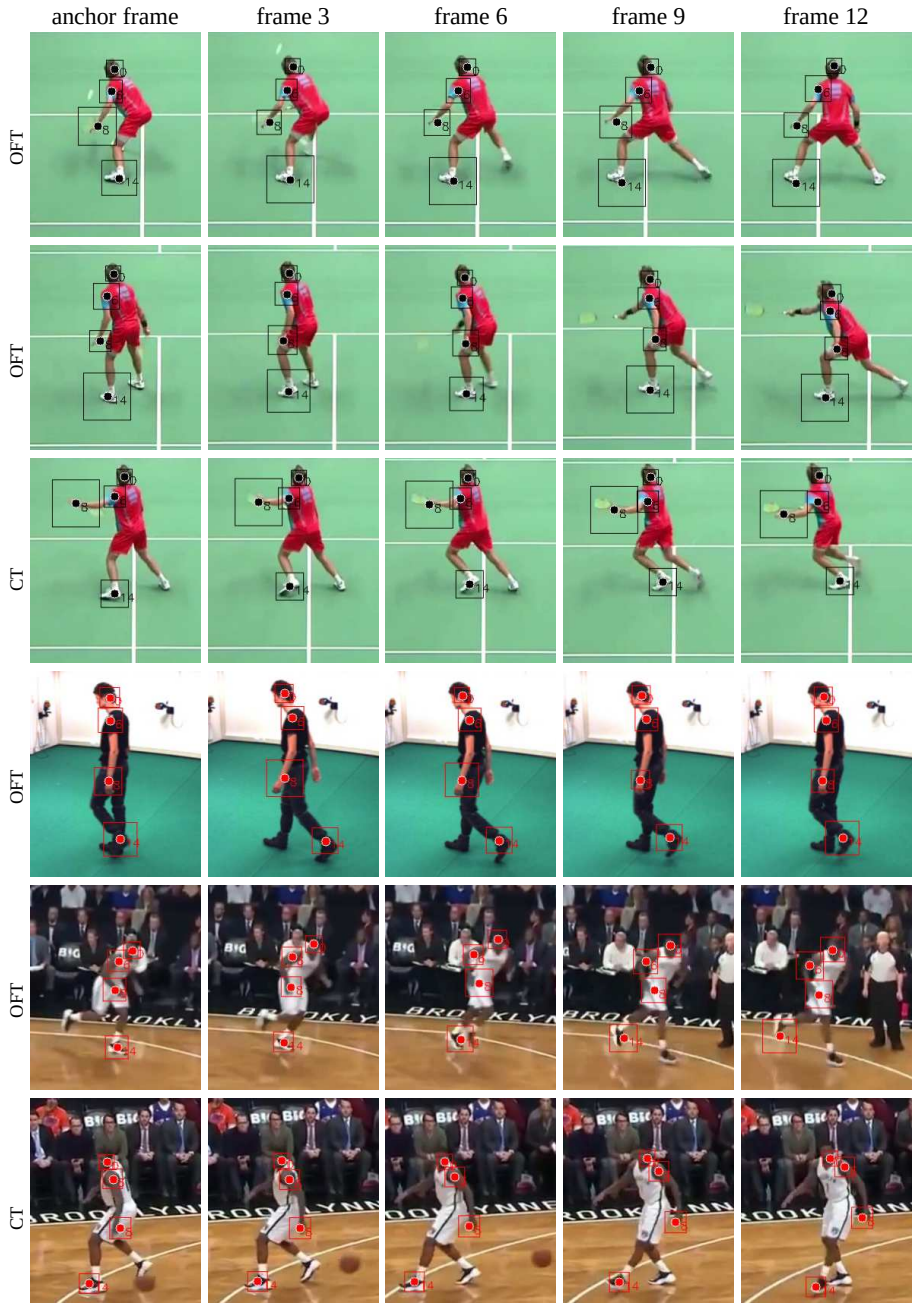


Fig. 5: **Qualitative OFT and CT landmark tracking results.** Results for three different YouTube badminton sequences (row 1-3), an UMPM sequence (row 4), and two different YouTube basketball sequences (row 6-7). Left column: manually annotated. All other columns: automatically annotated using landmark OFT and CT based tracking with automatic scale selection.