Follow that nose: tracking faces based on the nose region and image quality feedback

Luan P. Silva, Flávio H. de B. Zavan, Olga R. P. Bellon and Luciano Silva
IMAGO Research Group - Universidade Federal do Paraná
{luan.porfirio,flavio,olga,luciano}@ufpr.br

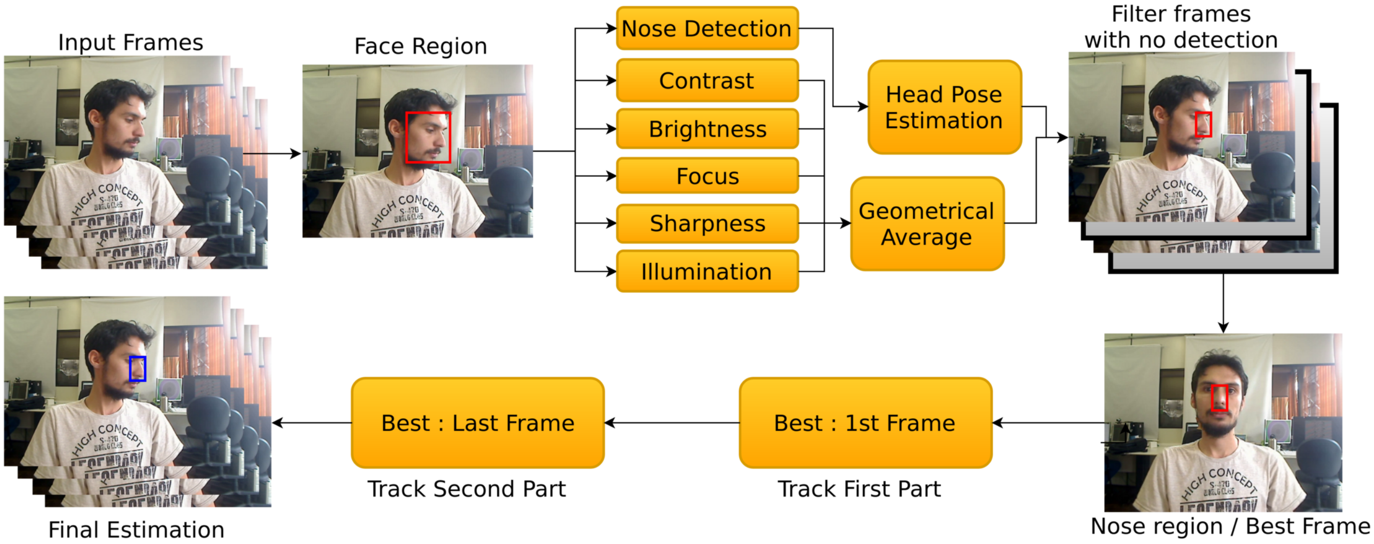

Fig. 1. Image quality and nose tracker diagram. Red lines are detected areas, blue and green areas are best and worst tracking predictions, respectively.

***Abstract*—Face tracking uses temporal information to infer the position of the face in each frame. One of its applications is in unconstrained (in-the-wild) environments where face detection methods fail to perform robustly. Current approaches presented in the literature are based on facial landmarks. Therefore, they have limitations when applied in in-the-wild environments as estimating the landmarks in such scenarios is not trivial. To address this issue, we propose a novel landmark-free approach based on a state-of-the-art generic visual tracking method, as baseline, combined with face quality assessment for initializing the tracking. In addition, we introduce using only the nose region as a solution for in-the-wild face tracking, initializing it with the nose of the best quality face in the video sequence. The nose is detected and used to estimate the head pose, which is combined with the face quality score for choosing the initialization frame. The nose region, rather than the entire face was chosen due to it being unlikely to be occluded, mostly invariant to facial expressions and visible in a long range of head poses. We performed experiments on the 300 Videos in the Wild dataset and our results favorably compared against the baseline method.**

***Keywords*-Face tracker; Nose region; Facial image quality**

I. Introduction

According to Jain *et al.* [1], facial recognition is one of the fundamental problems in computer vision. To this end, existing approaches assume that the face is detected on the first stage of the recognition pipeline. When working with videos, a successful face detection is not required for every frame, as the available temporal information allows the use of a facial tracker for increased face localization accuracy.

Because most state-of-the-art face tracking methods use landmarks [2], [3], they tend to not be robust in in-the-wild scenarios, including profile head poses where half of the landmarks are occluded by the face. Generic visual trackers provide landmark-free alternatives and have been successfully applied to predicting the location of a large range of objects, including faces [4], [5], [6], [7].

We propose the adaption of a generic visual tracker [4] for in-the-wild face tracking while using only to nose region for increased reliability. The nose has been shown to be efficient for biometrics [8], [9], it is visible even on profile faces, it is not easily deformed by expression and it is not likely to be occluded by accessories. We also explore a more extensive use of face detection [10] combined with face quality estimation [11], such that tracking can be performed both forwards and backwards starting from the highest quality frame (Figure 1). To evaluate our proposition, we perform tests on the 300 Videos in the Wild (VW) [12] and compare our results against the same tracker using the whole face and without the initial frame selection step.

This paper is organized as follows: existing related works are reviewed in Section 2; all steps of our approach are described in detail in Section 3; our results are presented and discussed in Section 4; and Section 5 includes final remakrs and a few words on future work.

II. Related work

Tracking methods are commonly subdivided into rigid and non-rigid approaches. Face tracking is usually done through non-rigid methods using landmarks [2], [3], [13], however profile poses are still challenging when tracking landmarks [14]. Rigid face tracking is similar to generic visual tracking, a bounding box of a single object in the first image is given and the position of the box is calculated in the subsequent frames, adapting to the changes in appearence of the object [4], [5], [6], [7].

Visual tracking algorithms are subdivided into two categories, generative and discrimininative trackers. The first uses generative models to find probable candidates for the next object location and are mostly based on principal component analysis (PCA) [15] or sparse representations [16], [17]. The latter learns binary classifiers to segment the target and the background, these can be based on Haar-like features [18], [19], [20], boosting variants [21], [22] and correlation filters [23], [24], but tend to have poor performance under uncontrolled environments, such as illumination changes, deformation and partial occlusion.

Recently, Convolutional Neural Networks (CNN) have received significant attention with state-of-the-art results on computer vision tasks such as image classification [25], object recognition [26], detection and segmentation [27]. Wang *et al.* [28] proposes a novel structured output CNN which transfers generic object features for online tracking, a CNN is pre-trained to distinguish objects from non-objects, and the output is a pixel-wise map that indicates the probability of each pixel belonging to the target. Hong *et al.* [7] learn a target-specific saliency map using a pre-trained CNN. Li *et al.* [29] learn two CNN classifiers from binary samples and do not require a pre-training procedure. The aforementioned CNN methods rely on positive and negatives training samples and only exploit features from the last CNN layer. According to Ma *et al.* [5] this information is insufficient for capturing spatial details such as target position, and fails on visual tracking approaches. Features from multiple convolutional layers have also been extracted [5], [6] on pre-trained networks on large-scale datasets such as VGG-Nets [30], [31].

Nam & Han [4] propose a MDNet (Multi-Domain Network), which learns from a set of videos with ground-truth annotations, achieving state-of-the-art results on the Visual Object Tracking Challenge [32].

III. Follow That Nose Approach

We adopt a Multi-Domain Network (MDNet) [4] as our base nose tracker. To increase tracking accuracy, we use the face quality information [11] combined with the head pose [33] to choose the best frame to initialize our tracker, performing tracking both forwards and backwards using the best frame as a starting point.

A. Multi-Domain Network (MDNet)

Existing convolutional neural network architectures for visual tracking [5], [6] are substantially smaller than the ones commonly used for typical recognition tasks such as AlexNet [25] and VGG-Nets [30], [31]. This is due to two different reasons: visual tracking distinguishes between only two classes, target and background; and deep CNNs are less effective for precise target localization as the deeper the network is, the more diluted the spatial information is [7]. MDNet-based trackers [4] are designed to learn shared features and classifiers specific to different tracking sequences.

During training, a generic representation is created in the shared layers across all domains. During testing, when evaluating a new sequence, a new branch is built from the initial frame using 500 positive and 5,000 negative samples around the ground-truth. For each subsequent frame, an online update is performed for increased robustness.

B. Face Quality Assessment

To assess the face quality, we first detect it using Faster R-CNN [10]. We trained our detector using the whole training subset (12,754 faces) from the Janus CS2 dataset[1]. The score is set to zero for the frames with no face detections. Abaza's *et al.*'s face quality estimation method [11] is then applied to the face region, generating scores for contrast, brightness, focus, sharpness and illumination, which are aggregated by calculating their geometrical average.

The nose region is then detected inside the face, using the same method. We trained our nose detector on a manually annotated subset (containing 6,435 noses) [33] of the IJB-A dataset [34], which already is a subset of Janus CS2. Head yaw pose information is extracted using the NosePose method [33] trained on the same subset.

The pose and quality information are combined such that face regions that did not yield any nose detections or have a near profile head yaw have their quality scores set to zero. The highest quality face's nose is used to initialize the tracking stage. This provides the tracker with a high quality nose region that includes little to no background, favoring success. The face quality assessment is used only for choosing the best frame for initializing the tracker. The detection of face and nose region are not considered when tracking.

C. Nose Tracking

Nam and Han's [4] original MDNet training using generic videos is applied for tracking the nose. It is used twice, forwards and backwards in time using the best frame as starting point, the online training step is performed individually for each subsequence. For each subsequent frame, 256 samples are collected with varying translation and scale based on a Gaussian distribution with the mean as the previous location. These are evaluated by the network and given a score, the

[1] The Janus CS2 dataset is currently not publicly available.

average of the top 5 samples is computed and presented as the final prediction. After estimation the position of the nose in all frames, both forwards and backwards results are concatenated, generating the final estimation for the whole video. Our nose tracker is presented in Algorithm 1 in function `estimateNoseLocation` and the main steps are also shown as a diagram in Figure 1.

Algorithm 1 Follow That Nose Tracker. The trackFrame function is the application of the visual tracker

```
function ESTIMATENOSELOCATION($frames$)
    $loadNetworkArchitecture()$
    $loadTrainedModel()$
    $bestFrameIndex, nose \leftarrow getBestNose(frames)$
    $initializeTracking(frames[bestFrameIndex])$
    $noses[bestFrameIndex] \leftarrow nose$
    for $i \leftarrow bestFrameIndex - 1$ down to $0$ do
        $noses[i] \leftarrow trackFrame(frames[i], noses[i+1])$
    end for
    for $i \leftarrow bestFrameIndex + 1$ to $len(frames) - 1$ do
        $noses[i] \leftarrow trackFrame(frames[i], noses[i-1])$
    end for
    return $noses$
end function
```

IV. Experimental Results

We quantitatively assess our method's performance against the baseline on the 300 Videos in the Wild dataset. It includes 114 challenging publicly available videos with a total of 222,093 frames. Every frame is annotated with 68 face landmarks, which were used for generating both the nose and face ground-truth when evaluating our and the baseline's performance on all videos in the dataset.

The baseline method is initialized with the ground-truth face annotation on the first frame of each video. Our method makes use of the procedure described in Section III-B to select the initial frame while using the detected nose region for performing the quality assessment. However, to achieve a fair comparison and not be affected by the nose region size differences in different datasets, the tracker is initialized using the ground-truth nose region.

The tracking accuracy is evaluated on a frame-by-frame basis, using two different metrics, the intersection coefficient [35] and the precision [32] (Algorithm 2). The increase in accuracy is clear when only the nose region is tracked, the predicted regions are calculated with greater precision when using the nose (Figure 3), however they do not intersect as tightly with the ground-truth when compared to the baseline (Figure 4). Figure 2 shows an example of the nose region being tracked with greater accuracy.

V. Conclusion

We presented an alternative approach for face tracking, integrating face quality assessment and using only the nose region. Given a video sequence, the initial tracking frame is

Algorithm 2 Both prediction evaluation metrics: Intersection Coefficient and Precision. The predicted region is represented as pred and the ground-truth region as gt

```
function ICOEFFICIENT($pred, gt$)
    $intersection \leftarrow getIntersection(pred, gt)$
    $iArea \leftarrow intersection.width * intersection.height$
    $pArea \leftarrow pred.width * pred.height$
    $gArea \leftarrow gt.width * gt.height$
    return $min(iArea/pArea, iArea/gArea)$
end function
function PRECISION($pred, gt$)
    return $l2norm(center(pred), center(gt))$
end function
```

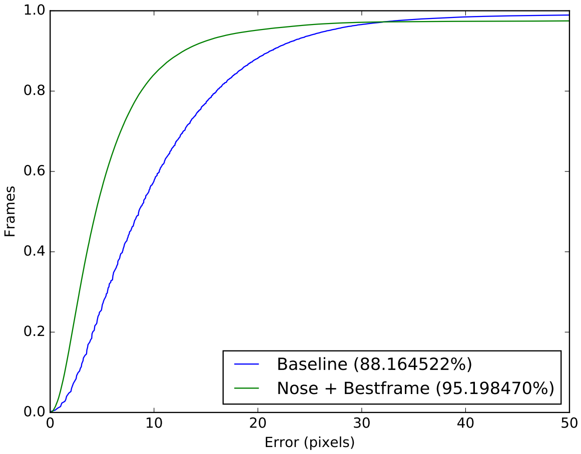

Fig. 3. Comparison of the precision metric between the baseline and our approach. The percentage of frames where the error was less than or equal to 20 pixels is displayed in parenthesis

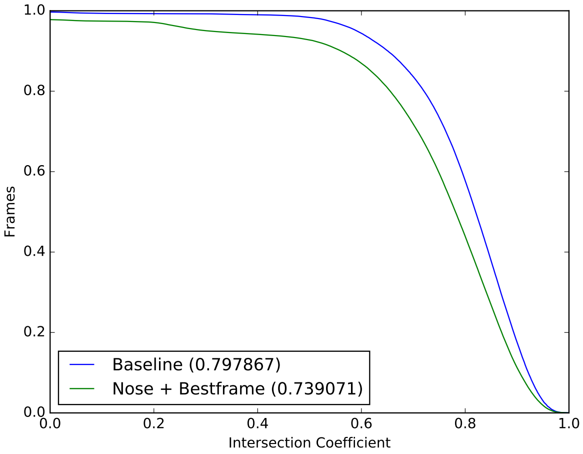

Fig. 4. Comparison of the intersection coefficient metric between the baseline and our approach. The area under the curve is displayed in parenthesis

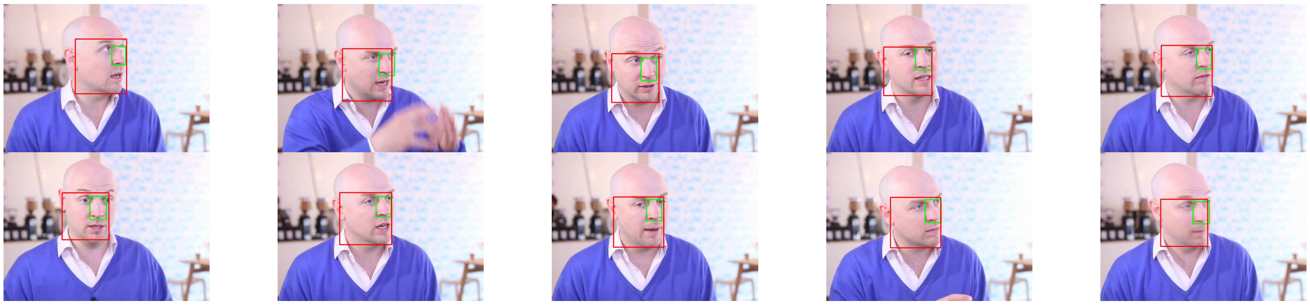

Fig. 2. Example sequence where the nose tracking (in green) performs better than the face (in red)

selected based on the image quality of the face and the head yaw. Tracking is performed on the nose, as it is consistent even in challenging environments. Our approach was applied to a state-of-the-art visual tracking method [4] and compared against the baseline, using the face region and initializing with the first frame. Experiments on the 300VW dataset [12] showed that we able to achieve higher precision rates in challenging scenarios. As part of future work, we would like to train the visual tracker using only nose images to increase the intersection performance and evaluate our approach on different datasets.

Acknowledgment

The authors would like to thank CNPq and CAPES for supporting this research.

References

[1] A. Jain, P. Flynn, and A. A. Ross, *Handbook of biometrics*. Springer Science & Business Media, 2007.

[2] S. Xiao, S. Yan, and A. Kassim, "Facial landmark detection via progressive initialization," in *IEEE ICCV Workshop*, 2015, pp. 33–40.

[3] J. Yang, J. Deng, K. Zhang, and Q. Liu, "Facial shape tracking via spatio-temporal cascade shape regression," in *IEEE ICCV Workshop*, 2015, pp. 41–49.

[4] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," in *IEEE CVPR*, 2016.

[5] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, "Hierarchical convolutional features for visual tracking," in *IEEE ICCV*, 2015.

[6] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *IEEE ICCV*, 2015, pp. 3119–3127.

[7] S. Hong, T. You, S. Kwak, and B. Han, "Online tracking by learning discriminative saliency map with convolutional neural network," in *ICML*, 2015.

[8] K. Chang, W. Bowyer, and P. Flynn, "Multiple nose region matching for 3d face recognition under varying facial expression," *IEEE TPAMI*, vol. 28, no. 10, pp. 1695–1700, 2006.

[9] N. Zehngut, F. Juefei-Xu, R. Bardia, D. K. Pal, C. Bhagavatula, and M. Savvides, "Investigating the feasibility of image-based nose biometrics," in *IEEE ICIP*, vol. 2, 2015.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *NIPS*. Curran Associates, Inc., 2015, pp. 91–99.

[11] A. Abaza, M. A. Harrison, T. Bourlai, and A. Ross, "Design and evaluation of photometric image quality measures for effective face recognition," *IET Biometrics*, vol. 3, no. 4, pp. 314–324, 2014.

[12] J. Shen, S. Zafeiriou, G. G. Chrysos, J. Kossaifi, G. Tzimiropoulos, and M. Pantic, "The first facial landmark tracking in-the-wild challenge: Benchmark and results," in *IEEE ICCV Workshop*, 2015, pp. 1003–1011.

[13] F. de la Torre, W.-S. Chu, X. Xiong, F. Vicente, X. Ding, and J. Cohn, "Intraface," in *IEEE FG*, vol. 1, 2015, pp. 1–8.

[14] G. G. Chrysos, E. Antonakos, P. Snape, A. Asthana, and S. Zafeiriou, "A comprehensive performance evaluation of deformable face tracking" in-the-wild"," *arXiv preprint arXiv:1603.06015*, 2016.

[15] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *IJCV*, vol. 77, no. 1-3, pp. 125–141, 2008.

[16] X. Mei and H. Ling, "Robust visual tracking using l1 minimization," in *IEEE ICCV*, 2009, pp. 1436–1443.

[17] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *IEEE CVPR*, 2012, pp. 2042–2049.

[18] K. Zhang, L. Zhang, and M.-H. Yang, "Real-time compressive tracking," in *ECCV*. Springer, 2012, pp. 864–877.

[19] S. Hare, A. Saffari, and P. H. Torr, "Struck: Structured output tracking with kernels," in *IEEE CVPR*, 2011, pp. 263–270.

[20] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[21] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting." in *BMVC*, vol. 1, no. 5, 2006, p. 6.

[22] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," in *ECCV*. Springer, 2008, pp. 234–247.

[23] M. Danelljan, G. Hager, F. Shahbaz Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *IEEE CVPR*, 2015, pp. 4310–4318.

[24] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE TPAMI*, vol. 37, no. 3, pp. 583–596, 2015.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1097–1105.

[26] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition." in *ICML*, 2014, pp. 647–655.

[27] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE CVPR*, 2014, pp. 580–587.

[28] N. Wang, S. Li, A. Gupta, and D.-Y. Yeung, "Transferring rich feature hierarchies for robust visual tracking," *arXiv preprint arXiv:1501.04587*, 2015.

[29] H. Li, Y. Li, and F. Porikli, "Deeptrack: Learning discriminative feature representations online for robust visual tracking," *IEEE Image Processing*, vol. 25, no. 4, pp. 1834–1848, 2016.

[30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[31] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014.

[32] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Cehovin, G. Fernandez, T. Vojir, G. Hager, G. Nebehay, and R. Pflugfelder, "The visual object tracking vot2015 challenge results," in *IEEE ICCV Workshops*, 2015, pp. 1–23.

[33] Authors, "Nosepose: a competitive, landmark-free methodology for head pose estimation in-the-wild," in *Submitted*, 2016.

[34] B. F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, M. Burge, and A. K. Jain, "Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a," in *IEEE CVPR*, 2015, pp. 1931–1939.

[35] A. Hoover, G. Jean-Baptiste, X. Jiang, P. J. Flynn, H. Bunke, D. B. Goldgof, K. Bowyer, D. W. Eggert, A. Fitzgibbon, and R. B. Fisher, "An experimental comparison of range image segmentation algorithms," *IEEE TPAMI*, vol. 18, no. 7, pp. 673–689, 1996.