Low-Quality Video Face Recognition with Deep Networks and Polygonal Chain Distance

Christian Herrmann*†, Dieter Willersinn†, Jürgen Beyerer†*
*Vision and Fusion Lab, Karlsruhe Institute of Technology KIT, Karlsruhe, Germany
†Fraunhofer IOSB, Karlsruhe, Germany
{christian.herrmann|dieter.willersinn|juergen.beyerer}@iosb.fraunhofer.de

***Abstract*—Face recognition under surveillance circumstances still poses a significant problem due to low data quality. Nevertheless, automatic analysis is highly desired for criminal investigations due to the growing amount of security cameras worldwide. We suggest a face recognition system addressing the typical issues such as motion blur, noise or compression artifacts to improve low-quality recognition rates. A low-resolution adapted residual neural net serves as face image descriptor. It is trained by quality adjusted public training data generated by data augmentation strategies such as motion blurring or adding compression artifacts. To further reduce noise effects, a noise resistant manifold-based face track descriptor using a polygonal chain is proposed. This leads to a performance improvement on in-the-wild surveillance data compared to conventional local feature approaches or the state-of-the-art high-resolution VGG-Face network.**

Fig. 1: Qualitative results of the proposed method on low-quality data. Line thickness and numbers denote face similarity (inverse of descriptor distance) and line color same (blue) and different (orange) identity.

I. Introduction

The increasing availability of security cameras raises the demand for analysis of the vast amounts of video footage, specifically, automatic analysis because manual inspection is unfeasible. While older security cameras lack in resolution and faces are often unrecognizable, with newer camera generations the faces become clearer and distinguishable. However, the data quality is usually still far from professional footage such as TV or press photographs, where automatic face recognition achieved impressive results recently, surpassing even human performance in certain setups [1, 2]. Addressing the low-quality surveillance domain is still a significant challenge for automatic face recognition approaches, caused by several reasons which are *misalignment*, *noise affection*, *lack of effective features* and *dimensional mismatch* between probe and gallery according to [3]. Recently, effective alignment methods for low-quality faces were proposed [4] which we found to be sufficiently accurate. Consequently, in this paper, we suggest an effective Convolutional Neural Network (CNN)-based feature which proves to be more efficient compared to previous solutions and address noise affection by data augmentation and a noise resistant track descriptor to utilize the temporal information. Data augmentation is necessary because large face datasets which are suitable for training a CNN are no surveillance datasets and consequently involve a domain gap. We suggest according augmentation strategies such as adding motion blur or compression artifacts to close this gap.

In detail, the contributions of this paper are threefold: First, the adaptation of the residual net architecture [5] to the low-quality face recognition domain by adjusting layer configuration and setup. Second, a manifold based and noise resistant strategy using a polygonal chain to aggregate facial information across multiple frames of a face track. Third, a systematic analysis of the target domain image quality effects and their reproduction as data augmentation for high-quality training data from a different domain.

II. Related Work

Addressing low-quality video face recognition involves several specific problems.

Face recognition with CNNs. Currently, CNNs serve only for high-resolution face recognition where they significantly improved the performance compared to previously known approaches, even surpassing human capabilities in certain setups [1, 2, 6]. Because these networks are mainly based on solutions for the ImageNet challenge [7], they adopt the high resolutions of 224×224 pixels and above. Low-resolution networks tend to loose performance as shown by [1], which makes it necessary to address this issue.

Low-quality face recognition. One part of the approaches addressing this task tries to mitigate the low data quality by preprocessing steps. This includes super-resolution methods where low-quality images are upsampled to apply a conventional high-resolution face-recognition strategy [8, 9] which proved to be a solid strategy for comparing low-quality

to high-quality gallery faces. For low-quality to low-quality matching feature adaptations [10], blur resistant features [11] or best-shot selection are known options. For video to video matching, preprocessing is usually too computationally expensive which is why we follow a different strategy and create a low-quality tolerant face descriptor.

Face track descriptors. Given a sequence of face image descriptors from one face track, there exists a wide variety of methods to aggregate these descriptors into a single track descriptor ranging from primitive best-shot selection to detailed manifold modeling. Usually, three categories are distinguished: set-based, space-based and manifold-based. Set-based methods include best-shot, random [2] or specific [12] selection of image descriptors, or even including all [13] image descriptors in the track descriptor. Set comparison can then be performed e.g. by minimum or Hausdorff distance [13]. Space-based modeling of sequences tries to fit a linear space model such as an affine subspace or the convex hull [14, 15]. Comparison can, e.g. , be performed by the principle angle between the subspaces [15]. In previous work, manifold-based methods operate directly on raw pixel values because it is known that the manifold assumption holds in this case [16]. It allows many possibilities to model the face sequence ranging from linear approximation by multiple PCA-planes [17] over simply applying locally linear embedding (LLE) [18] or combining LLE and k-means [16] to local probabilistic models [19]. Later on in section IV, we will motivate that the manifold assumption still holds in the case of CNN face descriptors under certain conditions and propose an appropriate model for this case. Especially, we take comparison time into consideration where some manifold methods lack in efficiency.

Recently successful alternative approaches (fitting in none of the categories) based on cumulative descriptors [20] are impractical in our case because they require local image features instead of the holistic ones produced by CNNs.

Dataset augmentation. To increase the generalization abilities of machine learning systems, especially across domains, data augmentation can be applied to the training data to increase the variety of data the system can learn. In the area of face recognition, this is no systematically studied topic yet. Parkhi et al. apply different crops and flipping to train their face network [6], but without any evaluation of the respective contribution. Their strategy is probably motivated by the common training strategies for the ImageNet challenge which apply cropping, flipping and color shift to improve the results [21, 22]. For low-quality data from surveillance, results for the person re-identification scenario indicate that different cropping, flipping and rotation are helpful, while color changes or affine transformations tend to decrease the results [23]. We will complement these augmentation suggestions with specific low-quality related effects such as motion blur or compression artifacts in section V. This way, the low-quality domain is directly addressed and the domain gap between training and test data is minimized.

TABLE I: Structure of the proposed low-quality residual network (LqNet). After each *conv* layer, batch normalization [24] is performed.

type	**size**	**stride, pad**	**data size out for 32×32 input image**
conv	3×3, 64	1, 1	32×32×64
max pool	3×3	2, 0	16×16×64
relu			16×16×64
res block (2×)	1×1, 64 3×3, 64 1×1, 256	2/1, 0 1, 1 1, 0	8×8×256
res block (2×)	1×1, 128 3×3, 128 1×1, 512	1, 0 1, 1 1, 0	8×8×512
res block (2×)	1×1, 256 3×3, 256 1×1, 1024	2/1, 0 1, 1 1, 0	4×4×1024
avg pool	3×3	1, 0	2×2×1024
fc			2048
relu			2048
fc			128

III. Face image description

To describe a given face track $T = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n]$, each frame $\boldsymbol{u}_i$ in the sequence is described by a CNN descriptor: $\boldsymbol{u}_i \mapsto \boldsymbol{z}_i$. Afterwards, the track descriptor presented in the next section compresses the information from several frames. A modified residual net architecture [5] adapted to the low-resolution verification task provides the face image descriptors. The low-quality residual network (LqNet) is trained from scratch because we found that fine-tuning pre-trained nets is unfeasible due to the large resolution difference between widely available high-resolution nets and the target scenario. We stack 3 double "bottleneck" building blocks as shown by table I. In addition, we found it beneficial to insert a fully connected layer between the last residual block and the output layer. Batch normalization [24] is applied after each convolutional layer. This results in a 20-layer architecture. By evaluation of further architectures with varying number of layers (15 to 50) this architecture was found to be the best solution.

For training the LqNet, a Siamese network structure [25] with max-margin loss l, instead of the contrastive loss, is applied. The loss function l is

$$l = \sum_{i,j} \max\left(0, 1 - y_{ij} \cdot \left(b - d^2(\boldsymbol{z}_i, \boldsymbol{z}_j)\right)\right), \tag{1}$$

similar to [26], where $\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ denote the face descriptors, $y_{ij} = \{-1, 1\}$ the indicator variable, b the decision boundary and d^2 the squared euclidean distance. In this way, the proposed network learns to map face images to discriminative 128-dimensional descriptors which can efficiently be compared by euclidean distance.

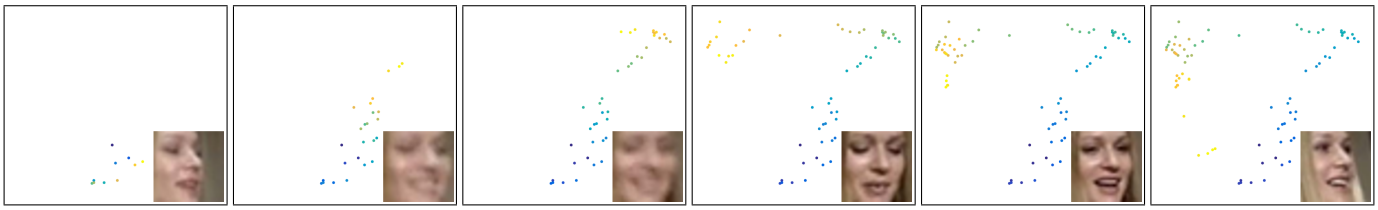
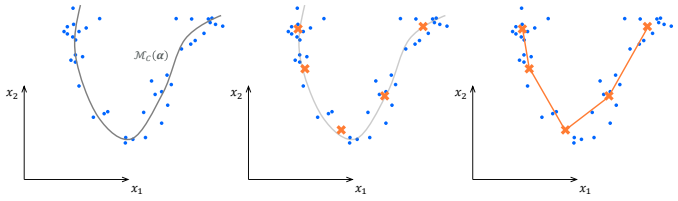


Fig. 2: Projection of face descriptors from a sample face sequence in 2D via PCA. Illustrated at different time steps with 10, 25, 40, 55, 70 and all (79) frames. Newly added descriptors are indicated in yellow-like colors and the face corresponding to the most recently added descriptor is indicated in the respective lower right corner. It can be observed that both principal axes of the descriptor space mainly correspond to two visible effects in this case: head rotation (vertical axis) and blur (horizontal axis).



(a) (b) (c)

Fig. 3: Visualization of the proposed manifold modeling. The face descriptors of one sequence (blue dots) are assumed to lie on a manifold $\mathcal{M}_\mathcal{C}$ (a). This manifold is first approximated by local means (b) and then linearly reconstructed as polygonal chain by the line segments between the means (c).

Fig. 4: Example shots from our recorded surveillance data.

IV. Track Representation

Using the net from the previous section yields one descriptor per image: $\boldsymbol{u}_i \mapsto \boldsymbol{z}_i$. To process face tracks as delivered by a face tracker, one requires a track descriptor $\mathcal{D}$ which models all image descriptors from one track: $[\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n] \mapsto \mathcal{D}$. Due to image noise potentially influencing single image descriptors, a robust track descriptor is required. When analyzing face descriptor sequences as in figure 2, it appears obvious that this sequence can be modeled by a manifold. Assuming a manifold is justified, because

- face images $\boldsymbol{u}$ reside on a manifold $\mathcal{M}$ when vectorized (refers to raw pixel values) [16],
- linear transformations usually preserve manifolds and the convolutional neural net $\mathcal{C}$ mainly consists of linear operations, thus leading to a transformed manifold $\mathcal{M}_\mathcal{C}$, and
- non-linearities are assumed to be noise $\boldsymbol{r}$: $\mathcal{D} = \mathcal{C}(\boldsymbol{u}) = \mathcal{M}_\mathcal{C}(\boldsymbol{\alpha}) + \boldsymbol{r}$ where $\boldsymbol{\alpha}$ denotes the manifold coordinate vector.

Starting from the manifold concept (figure 3a), a noise resistant representation is required. We propose k local means $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}$ to address this (figure 3b). In contradiction to more common exemplar based representations such as [12] or applying the Ramer-Douglas-Peucker algorithm to select good exemplars, the proposed strategy includes an averaging effect which reduces the given noise.

We then model the manifold as the polygonal chain given by connecting the local means $S_p = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k]$ (figure 3c). Let $S = [\boldsymbol{x}_1, \boldsymbol{x}_2] \cup \ldots \cup [\boldsymbol{x}_{k-1}, \boldsymbol{x}_k]$ denote the point set of the manifold model and further s_i the line segment between the points $\boldsymbol{x_i}$ and $\boldsymbol{x_{i+1}}$ of the polygonal chain. Then the polygonal chain distance (PCD) $d(S^1, S^2)$ between two polygonal chains S^1 and S^2 is the minimum euclidean distance between any two points $\boldsymbol{t^1}, \boldsymbol{t^2}$ on the chains which is equivalent to the minimum distance between any two line segments:

$$d(S^1, S^2) = \min_{\substack{\boldsymbol{t}^1 \in S^1, \\ \boldsymbol{t}^2 \in S^2}} d\left(\boldsymbol{t}^1, \boldsymbol{t}^2\right) = \min_{\substack{i \in \{1, \ldots, k^1 - 1\}, \\ j \in \{1, \ldots, k^2 - 1\}}} d(s_i^1, s_j^2). \quad (2)$$

Segment to segment distance $d(s_i^1, s_j^2)$ in $\mathbb{R}^n$ is non-trivial and basically three cases can be distinguished. For details how to determine when exactly to apply which case refer e.g. to [27].

1) Closest points of according lines lie inside the segments. In this case line to line distance applies:

$$d(s_i^1, s_j^2) = \left\| \boldsymbol{a}_{ij} - \left(\boldsymbol{a}_{ij}^T \boldsymbol{g}_i^1\right) \cdot \boldsymbol{g}_i^1 - \left(\boldsymbol{a}_{ij}^T \boldsymbol{g}_j^2\right) \cdot \boldsymbol{g}_j^2 \right\|_2 \quad (3)$$

with

$$\boldsymbol{a}_{ij} = \boldsymbol{x}_i^1 - \boldsymbol{x}_j^2 \quad (4)$$

and

$$\boldsymbol{g}_i^k = \frac{\boldsymbol{x}_{i+1}^k - \boldsymbol{x}_i^k}{\left\| \boldsymbol{x}_{i+1}^k - \boldsymbol{x}_i^k \right\|_2} \quad (5)$$

denoting the normalized directional vector of the respective line.

2) Point to segment distance applies if the endpoint of one segment is part of the minimum distance. For endpoint $\boldsymbol{x}_i^1$ w.l.o.g. the distance is

$$d(s_i^1, s_j^2) = \left\| \boldsymbol{a}_{ij} - \left(\boldsymbol{a}_{ij}^T \boldsymbol{g}_j^2\right) \cdot \boldsymbol{g}_j^2 \right\|_2. \quad (6)$$

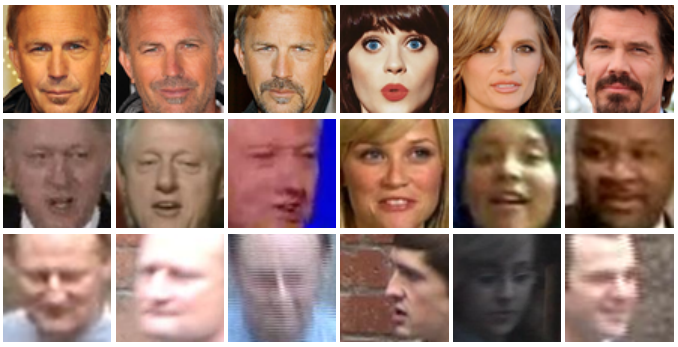


Fig. 5: Qualitative comparison of face images from FaceScrub (top), YTF (mid) and SURV (bottom). It can be seen that video data (YTF and SURV) has less quality than single image, mugshot-like data (FaceScrub). However, professional video recordings (YTF) show still better quality than surveillance data (SURV).

3) Point to point distance is the right choice if the endpoints of both segments are part of the minimum distance.

$$d(s_i^1, s_j^2) = \min_{\substack{u\in\{i,i+1\},\\ v\in\{j,j+1\}}} \left\| \boldsymbol{x}_u^1 - \boldsymbol{x}_v^2 \right\|. \quad (7)$$

V. Data and Augmentation

To represent the target scenario, an in-the-wild surveillance dataset (figure 4) was recorded on different days and nights at different locations, including indoor and outdoor, with several cameras per location. Faces are tracked by a Viola-Jones based face tracker and a subset of all detected face tracks was labeled resulting in a dataset size of 869 face tracks of 25 people. Face sizes are mostly in the range of 20 to 40 pixels and the track length varies from 14 to about 1,200 frames with an average length of 59 frames. For further references let's call this dataset SURV.

Due to the limited size of the collected data, it is unfeasible to train a neural network with a subset of this data. It is also economically unreasonable to try creating a sufficiently large surveillance training dataset because of the required manual labeling. Thus, training with large public face datasets is required. The problem consists in the domain gap between these datasets and the surveillance domain, because they are mainly automatically collected high-quality celebrity face images from the web (e.g. Celebrity-1000 [28], FaceScrub [29], MS-Celeb-1M (MS1M) [30], MSRA-CFW (MSRA) [31], YTF [32]).

Two strategies lead to target domain adaptation. First, in addition to public datasets, we add a TV Collection (TVC) face video dataset (15.4K tracks, 604 persons). Although collected from professional TV footage, this video data is closer to the target domain than public single image datasets. It has similar image quality as the YTF or Celebrity-1000 dataset, but is significantly larger than the first and has less label errors than the second. Second, we are looking for image transformations that adjust the public high-quality datasets to be similar to the low-quality target domain with respect to image properties. Obviously, the first stage is required to be an adjustment to the chosen target resolution of 32×32 pixels face size. The next domain difference is blur. Comparing SURV and downsampled versions of FaceScrub and YTF with regard to blur by the maximum response of a Laplacian filter indicates the respective sharpness level. High responses of the Laplacian filter denote sharp edges which are typical for unblurred images. Thus, filter responses can be understood as sharpness level. FaceScrub images show an average sharpness level of 0.534 and YTF images 0.352, compared to 0.300 for SURV face images. This correlates well with the subjective image quality as shown by figure 5. According to further tests, the difference corresponds on average to blurring the FaceScrub images with a Gaussian kernel with $\sigma = 0.6$ or a motion kernel of 5 pixels length ($\sigma = 0.4$ and 1.5 pixels for YTF). Note that blur in surveillance data is usually motion blur caused by object movement and integration time of the camera. The image formation process involves some further effects besides motion blur and scale effects including noisy images caused by sensor quantum noise as well as artifacts caused by compression requirements to transmit the data.

TABLE II: Training data augmentation strategies with their application probabilities and intensity. Unless normal distribution $\mathcal{N}$ is denoted, intensity is uniformly distributed over the range. Given parameters assume $[0, 1]$ pixel value range. m denotes the number of active domain augmentations when combining several augmentations strategies.

augmentation	probability	intensity
crop	0.8	up to 2 pixels
flip	0.5	
rotation	0.5	$2{\cdot}\mathcal{N}(0,1)$ degrees
motion blur	$\frac{0.5}{m}$	up to 5 pixels
noise	$\frac{0.5}{m}$	up to $0.1{\cdot}\mathcal{N}(0,1)$
compression	$\frac{0.5}{m}$	jpeg quality down to 6
rescale	$\frac{0.5}{m}$	up to factor 1.4

All in all, this leads to seven different augmentation strategies, the first geometric three inspired by literature [6, 23] and the last four by the domain requirements: flipping, cropping, rotation, motion blur, noise, compression and rescaling. For a training sample each augmentation is applied at runtime with a certain predefined probability and a bounded random effectiveness presented in detail by table II which reflect the frequency and intensity in the target domain.

VI. Experiments

First, we shed some light on the process to train the proposed LqNet face image descriptor. This includes the choice of appropriate training data and augmentation strategies. Second, some insight for the PCD track descriptor is given. Finally, a comparison to state-of-the-art face recognition approaches on the target domain is performed. All results in this section are based on a 10-fold cross validation verification setup. Values are given as accuracy with its standard deviation (std) across

TABLE III: Comparison of several datasets and their suitability to train the proposed LqNet face descriptor.

train dataset	public	augmentation	
		no	yes
		accuracy±std	accuracy±std
Celebrity-1000 [28]	yes	0.642±0.005	0.651±0.003
FaceScrub [29]	yes	0.621±0.005	0.641±0.002
MS1M [30]	yes	0.600±0.004	0.610±0.006
MSRA [31]	yes	0.538±0.006	0.603±0.004
TVC	no	0.668±0.007	0.686±0.004

TABLE IV: Impact of the geometric (*) and quality augmentation strategies on validation results when training only with the FaceScrub dataset.

augmentation	none +	geometric +
	accuracy±std	accuracy±std
none	0.621±0.005	0.609±0.005
* crop	0.638±0.003	-
* flip	0.628±0.003	-
* rotation	0.617±0.003	-
motion blur	0.641±0.002	0.602±0.004
noise	0.633±0.004	0.639±0.005
compression	0.628±0.004	0.626±0.005
rescale	0.621±0.006	0.630±0.005

the folds and complemented by area under curve (AUC) and equal error rate (EER) in the final experiments. AUC and EER denote different characteristics of the ROC-curve: AUC denotes the area under the ROC curve, while the EER denotes the point where both errors namely the false positive rate and the false negative rate are equal.

A. LqNet face descriptor

Optimization of the LqNet is performed with a validation dataset consisting of single face images from the target domain. The nets are trained for a fixed number of iterations for fair comparison using the Caffe framework [33] on a GeForce Titan X. First, possible training datasets are checked for their suitability with respect to the target domain using no augmentation. As table III indicates, the best results are achieved with the video datasets Celebrity-1000 [28] and our own TVC caused by the smaller domain gap to the target domain. Both Microsoft datasets yield lower results than the rest, probably caused by rather inaccurate automatic labeling. For each dataset, an analysis regarding useful augmentation strategies to close the respective gap to the target domain is performed. Table IV shows the augmentation results for the FaceScrub dataset. In this case, motion blur, noise and different crops improve performance the most. The best results when using augmentation strategies are listed in table III for each dataset. Supported by the results in tables III and IV, the final configuration is the training on the datasets Celebrity-1000, FaceScrub and TVC with crop, flip, motion blur and noise augmentations. Consequently, the training set contains about 3.3M face images from 1,930 persons. Using this setup, an accuracy of 0.691 ± 0.006 is achieved on the validation set after the same number of iterations as before and 0.756 ± 0.004 after convergence.

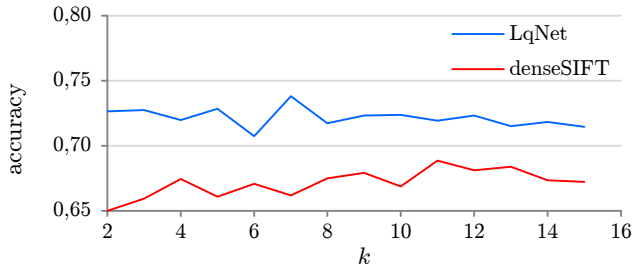


Fig. 6: Influence of local patch number k for the proposed PCD comparison method. Evaluated on SURV dataset.

B. PCD track descriptor

The proposed track descriptor requires the choice of the segment number k. Figure 6 indicates that performance is changing only slowly for changes in k, making the choice robust. The trend shows a decrease in accuracy for many local means k when using the LqNet descriptor which can be expected because of the decreasing averaging effect. This matches the desire to keep k small because track comparison complexity grows with $O(k^2)$. For further experiments we use $k = 2$ or $k = 5$.

C. Comparison to other approaches

Regarding experiments on the target domain, the proposed LqNet face image descriptor is compared to raw pixel values, local features (LBP and dense SIFT) and the state-of-the-art VGG-Face network [6]. As vector distance for the image descriptors, the best one out of euclidean, cosine and Hellinger is chosen for each face image descriptor. Two set-based methods, namely best-shot and minset, the space-based method MSM [15] and the manifold-based LLE [18] serve as comparative face track descriptors. The results for all image and track descriptor combinations are listed in table V and indicate mainly three aspects. First, for CNN-based image descriptors, the proposed PCD track descriptor yields superior results compared to the other options. Second, while the VGG-Face descriptor unexpectedly yields the worst results of all image descriptors, the proposed LqNet descriptor outmatches its rivals due to addressing the domain gap between training and target data. Third, the advantage of CNN-based descriptors compared to local features (LBP and dense SIFT) is far lower for the low-quality domain than for the high-quality domain. Comparing the approaches on the high-quality official YTF evaluation setup with 32×32 pixels downscaled faces (table VI) shows a significantly larger gap in this case than for the low-quality case in table V. CNN-based approaches show a significant loss in performance for low-quality data, especially the VGG-Face network suffers heavily under the low-quality conditions. The proposed LqNet still looses some of its advance compared to local features on low-quality data, nevertheless its performance remains better than these baseline approaches.

TABLE V: Comparison of the proposed LqNet face descriptor and the proposed PCD track descriptor with further descriptors on the SURV dataset at 32×32 pixels face size. For each face descriptor, the applied vector distance is denoted. For details refer to text.

track distance	**value**	***LqNet* *(euclidean)***	**VGG-Net [6] (cosine)**	**dense SIFT [26] (Hellinger)**	**LBP [34] (Hellinger)**	**raw pixel (cosine)**
best shot	accuracy±std	0.640±0.084	0.571±0.079	0.572±0.065	0.551±0.063	0.549±0.046
	AUC | EER	0.697 | 0.410	0.585 | 0.438	0.596 | 0.437	0.600 | 0.443	0.562 | 0.454
minset	accuracy±std	0.709±0.102	0.609±0.093	**0.700±0.035**	**0.698±0.072**	**0.639±0.079**
	AUC | EER	0.799 | 0.336	0.653 | 0.395	0.750 | 0.309	0.734 | 0.328	0.699 | 0.337
MSM	accuracy±std	0.705±0.126	0.617±0.097	0.644±0.043	0.651±0.053	0.633±0.086
	AUC | EER	0.826 | 0.315	0.679 | 0.369	0.701 | 0.351	0.698 | 0.358	0.673 | 0.367
LLE	accuracy±std	0.615±0.071	0.557±0.028	0.525±0.039	0.538±0.043	**0.639±0.101**
	AUC | EER	0.656 | 0.458	0.562 | 0.458	0.539 | 0.468	0.542 | 0.470	0.686 | 0.357
PCD ($k = 2$)	accuracy±std	0.726±0.102	**0.628±0.087**	0.650±0.034	0.646±0.055	0.626±0.070
	AUC | EER	0.809 | 0.288	0.674 | 0.370	0.713 | 0.347	0.695 | 0.349	0.697 | 0.351
PCD ($k = 5$)	accuracy±std	**0.728±0.091**	0.625±0.093	0.677±0.048	0.650±0.052	0.567±0.044
	AUC | EER	0.802 | 0.303	0.668 | 0.378	0.733 | 0.325	0.700 | 0.353	0.564 | 0.449

TABLE VI: Results for high-quality public YTF dataset at 32×32 pixels face size. For PCD $k = 2$.

method	**accuracy±std**
raw pixel (cosine) + minset	0.604±0.030
LBP (Hellinger) + minset	0.649±0.015
dense SIFT (Hellinger) + minset	0.664±0.013
VGG-Face (cosine) + *PCD*	0.854±0.012
LqNet (euclidean) + *PCD*	0.806±0.017

VII. Conclusion

As the last experiments showed, low-quality face recognition is an even harder task for current face recognition approaches than low-resolution alone already is. To improve low-quality performance, the proposed face image descriptor has proved to be a valid solution when combined with motion blur and noise data augmentation strategies for training data. In addition, the presented manifold-based face track descriptor based on a polygonal chain improved performance for CNN-based descriptors with its integrated averaging character. All in all, low-quality face recognition appears to benefit from the current progress in the high-quality domain if accompanied by the according transfer strategies. Nevertheless, we think that this topic remains a challenging field for research.

References

[1] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," in *Computer Vision and Pattern Recognition*, 2015, pp. 815–823.

[2] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708.

[3] Z. Wang, Z. Miao, Q. M. J. Wu, Y. Wan, and Z. Tang, "Low-resolution face recognition: a review," *The Visual Computer*, vol. 30, no. 4, pp. 359–386, 2014.

[4] C. Qu, E. Monari, and T. Schuchert, "Resolution-aware Constrained Local Model with mixture of local experts," in *Advanced Video and Signal Based Surveillance Workshops*, 2013, pp. 454–459.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[6] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," *British Machine Vision Conference*, vol. 1, no. 3, p. 6, 2015.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 248–255.

[8] X. Wang and X. Tang, "Hallucinating face by eigentransformation," *Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 35, no. 3, pp. 425–434, Aug 2005.

[9] P. H. Hennings-Yeomans, S. Baker, and B. V. Kumar, "Simultaneous super-resolution and feature extraction for recognition of low-resolution faces," in *Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[10] C. Herrmann, "Extending a local matching face recognition approach to low-resolution video," in *Advanced Video and Signal Based Surveillance*, 2013.

[11] V. Ojansivu and J. Heikkilä, "Blur Insensitive Texture Classification Using Local Phase Quantization," in *Image and Signal Processing*. Springer, 2008, pp. 236–243.

[12] M. Zhao, J. Yagnik, H. Adam, and D. Bau, "Large Scale Learning and Recognition Of Faces in Web Videos," in *Automatic Face and Gesture Recognition*, 2008, pp. 1–7.

[13] S. Chen, S. Mau, M. T. Harandi, C. Sanderson, A. Bigdeli, and B. C. Lovell, "Face Recognition from Still Images to Video Sequences: A Local-feature-based Framework," *EURASIP Journal on Image and Video Processing*, 2011.

[14] H. Cevikalp and B. Triggs, "Face recognition based on image sets," in *Computer Vision and Pattern Recognition*, 2010.

[15] K. Fukui and O. Yamaguchi, "Face Recognition Using Multi-viewpoint Patterns for Robot Vision," *Robotics Research*, pp. 192–201, 2005.

[16] A. Hadid and M. Pietikainen, "From still image to video-based face recognition: an experimental analysis," in *Automatic Face and Gesture Recognition*. IEEE, 2004, pp. 813–818.

[17] K. Lee, J. Ho, M. Yang, and D. Kriegman, "Video-Based Face Recognition Using Probabilistic Appearance Manifolds," *Computer Vision and Pattern Recognition*, vol. 1, pp. 313–320, 2003.

[18] A. Hadid and M. Pietikäinen, "Manifold learning for video-to-video face recognition," *Biometric ID Management and Multimodal Communication*, pp. 9–16, 2009.

[19] M. E. Wibowo, D. Tjondronegoro, L. Zhang, and I. Himawan, "Heteroscedastic probabilistic linear discriminant analysis for

manifold learning in video-based face recognition," in *Workshop on Applications of Computer Vision*, 2013, pp. 46–52.
[20] O. M. Parkhi, K. Simonyan, A. Vedaldi, and A. Zisserman, "A Compact and Discriminative Face Track Descriptor," in *Computer Vision and Pattern Recognition*, 2014.
[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.
[23] N. McLaughlin, J. M. Del Rincon, and P. Miller, "Data-augmentation for reducing dataset bias in person re-identification," in *Advanced Video and Signal Based Surveillance*. IEEE, 2015, pp. 1–6.
[24] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015.
[25] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2006, pp. 1735–1742.
[26] K. Simonyan, O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Fisher vector faces in the wild," in *British Machine Vision Conference*, vol. 1, no. 2, 2013, p. 7.
[27] D. Eberly, *3D Game Engine Design: A Practical Approach to Real-Time Computer Graphics*. CRC Press, 2006.
[28] L. Liu, L. Zhang, H. Liu, and S. Yan, "Toward large-population face identification in unconstrained videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 11, pp. 1874–1884, 2014.
[29] H.-W. Ng and S. Winkler, "A data-driven approach to cleaning large face datasets," in *International Conference on Image Processing*. IEEE, 2014, pp. 343–347.
[30] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "MS-Celeb-1M: Challenge of Recognizing One Million Celebrities in the Real World," in *Imaging and Multimedia Analytics in a Web and Mobile World*, 2016.
[31] X. Zhang, L. Zhang, X.-J. Wang, and H.-Y. Shum, "Finding celebrities in billions of web images," *IEEE Transactions on Multimedia*, vol. 14, no. 4, pp. 995–1007, 2012.
[32] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Computer Vision and Pattern Recognition*, 2011.
[33] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding," *arXiv preprint arXiv:1408.5093*, 2014.
[34] T. Ahonen, A. Hadid, and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," *Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.