Boosting Face in Video Recognition via CNN based Key Frame Extraction

Xuan Qi, Chen Liu and Stephanie Schuckers
Clarkson University
8 Clarkson Ave., Potsdam, NY 13699, US
{qix,cliu,sschucke}@clarkson.edu

Abstract

Face in video recognition (FiVR) technology is widely applied in various fields such as video analytics and real-time video surveillance. However, FiVR technology also faces the challenges of high-volume video data, real-time processing requirement, as well as improving the performance of face recognition (FR) algorithms. To overcome these challenges, frame selection becomes a necessary and beneficial step before the FR stage. In this paper, we propose a CNN-based key-frame extraction (KFE) engine with GPU acceleration, employing our innovative Face Quality Assessment (FQA) module. For theoretical performance analysis of the KFE engine, we evaluated representative one-person video datasets such as PaSC, FiA and ChokePoint using ROC and DET curves. For performance analysis under practical scenario, we evaluated multi-person videos using ChokePoint dataset as well as in-house captured full-HD videos. The experimental results show that our KFE engine can dramatically reduce the data volume while improving the FR performance. In addition, our KFE engine can achieve higher than real-time performance with GPU acceleration in dealing with HD videos in real application scenarios.

1. Introduction

Face recognition (FR) is an essential tool for solving identification problems in video analytics and video surveillance. Previous researches using holistic methods such as Principal Component Analysis (PDA), Linear Discriminant Analysis (LDA) and Independent Component Analysis (ICA) have shown their capability for FR [11]. Later, researchers also exploited methods of extracting key components of human face to solve FR problem. Ahonen et al. proposed a method utilizing Local Binary Pattern (LBP) of facial texture [1]. Scale Invariant Feature Transform (SIFT) is another method proposed by Bicego et al. [5] which generates key-points of faces for matching purpose.

Different from extracting predefined types of face features, Deep Neural Network (DNN) based approaches have shown a much stronger and more robust capability and elevate FR performance to a whole new level. Taigman et al. [24] proposed their DeepID network and achieved near human-level FR performance. Parkhi et al. proposed VGG network trained with large face dataset [17]. In their experiment, the VGG network achieved a very high performance in Labeled Faces in the Wild (LFW) [10] and YouTube Faces in the Wild (YTF) [26] datasets. Similar works such as OpenFace [2], FaceNet [20] and DeepID [22] are also DNN-based approaches which achieved very promising FR performance. In real implementation of the neural network, the second-to-last fully-connected layer right before the final classification layer can be used as the low-dimension representation of faces [17, 2]. Then, the FR or face matching between a query face image and face image gallery is completed by comparing their euclidean distance or cosine distance. Apparently, one benefit by following this approach is that we can perform FR on newly added identities without retraining the DNN. Hence, we use this framework as FR back-end in this work.

1.1. FiVR using DNN Based Feature Extractor

A typical DNN-based FiVR system can be described as in Figure 1. For recognizing a person's face image extracted from video sequences, we need to prepare a gallery set which consists of still face images beforehand. For a query, we use a feature extractor to transform the query images and gallery images into feature space. Traditionally, the features in feature space can be predefined as LBP, HAAR or HOG features [29]. With the development of deep learning technology, the deep neural network (DNN) such as VGG or GoogLenet [23] can also be used for image feature extraction. In this work, we use VGG network trained with VGG face database and GoogLenet trained with CASIA-WebFace dataset as feature extractors. The feature for query image and gallery images generated by DNN module is a 1-D "deep feature vector". Then, the comparison between query image and galley is transferred to the comparison between feature vector of query image and the vector gallery

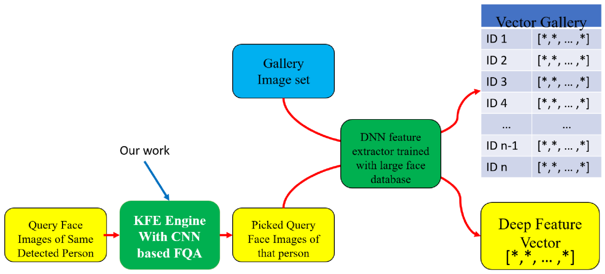

Vector Gallery
ID 1 [*,*, ... ,*]
ID 2 [*,*, ... ,*]
ID 3 [*,*, ... ,*]
ID 4 [*,*, ... ,*]
... ...
ID n-1 [*,*, ... ,*]
ID n [*,*, ... ,*]
Gallery Image set
DNN feature extractor trained with large face database
Our work
Query Face Images of Same Detected Person
KFE Engine With CNN based FQA
Picked Query Face Images of that person
Deep Feature Vector [*,*, ... ,*]

Figure 1: Typical FiVR framework

list generated from image gallery set. **[R3A3] The one with the nearest distance to the query sample from the galley will be picked as a match. However, there exists a chance that the nearest one maybe the wrong identity. This is inherited from the FR back-end itself. As a result, in this work when we evaluate the FR performance, we do examine the correctness.**

1.2. Our Proposed Approach

As shown in Figure 1, what we propose is a key-frame extraction (KFE) engine with CNN-based Face Quality Assessment (FQA) module, which lies between the face detection/tracking and face recognition (FR) procedures. The function of our KFE engine is to pick frames with the best quality face images of the detected persons and forward them to FR back-end. The benefits of using KFE engine come in two folds: first, we can reduce the data volume to be forwarded to FR backend, which dramatically reduces the computing overhead for performing FR, especially DNN-based FR. Second, human faces in videos alway come with various head pose changes and background condition changes such as lighting and appearance of other objects. Hence, the KFE engine has the potential to improve FR performance by rejecting video frames with poor quality faces which can cause wrong identification result.

Figure 2 shows the structure of our KFE module. For all incoming video frames, we perform face detection and tracking first. Then, for detected faces we perform Face Quality Assessment (FQA) and extract key-frames with best quality face for each identity. To emphasize, face detection, face tracking and CNN FQA module in our KFE engine are all accelerated by GPU. **[R1A2] In our implementation, if the number of people across frames changes, we deem it a scene change and the KFE engine will forward the current key-frame(s) to the back-end FR engine, then flush the key-frame buffer and begin to generate new key-frames. Hence, the KFE engine could generate more than one key-frame for the same identity. [R1A3] Besides, in our framework we can set different threshold to readily pick top-N best frames instead of single best frame. By doing so, we can employ spatial and temporal methods to restore more details of the face image for a single identity to elevate the FR performance in future research.**

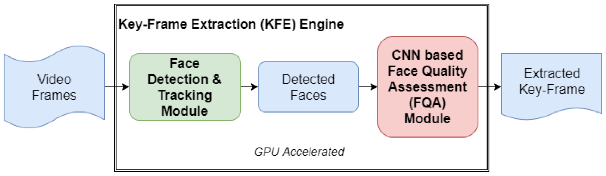

Key-Frame Extraction (KFE) Engine
Video Frames
Face Detection & Tracking Module
Detected Faces
CNN based Face Quality Assessment (FQA) Module
Extracted Key-Frame
GPU Accelerated

Figure 2: Structure of our KFE engine

Table 1: Typical Existing Face Database

Databases	#Identities	#Images	Public?
LFW [26]	5,749	13,233	Y
Celeb-Faces [21]	10,177	202,599	N
SFC [24]	4,030	4,400,000	N
VGG [17]	2,622	2,622,000	Y
CASIA-WebFace [28]	10,575	494,414	Y

At the FR back-end, in this work we applied cosine similarity as the FR performance value to generate final FR result. For a vector generate from query image $V(q)$, and a vector V(g_i) from the vector gallery G, the cosine similarity between them is described as in Equation 1. For all identities in gallery set, we can generate a list $\{$ ID_1: Sim(V(q),V(g_1)); ID_2: Sim(V(q),V(g_2)); ...; ID_{n-1}: Sim(V(q),V(g_{n-1})); ID_n: Sim(V(q),V(g_n))$\}$, which contains all FR performance values for all identities in the gallery. Then, the FR result from Rank 1 to Rank N can be generated by sorting the FR performance values from high to low for the list of identities. For constructing DNN feature extractor in FR back-end, we choose two datasets: VGG [17] and CASIA-WebFace [28] to train two different DNN feature extractors for different experimental purposes from existing face datasets listed in Table 1, according to their scale in identities, total image number and accessibility.

$$Sim(V(q), V(g_i)) = \frac{V(q) \cdot V(g_i)}{||V(q)|| \cdot ||V(g_i)||} \tag{1}$$

The rest of paper is organized as follows: in Section 2, we review several main-stream research in FQA-based KFE for FiVR. Section 3 describes design details and performance evaluation at the core of our KFE engine: the CNN based FQA module. Section 4 is the experiments we conducted to evaluate our approach. Lastly, we conclude in Section 5.

2. Related Works

The existing frame selection approaches for FiVR can be divided into three main categories [7]: face image clustering based, optical-flow based and face image quality based. The clustering approach treats video with people as a set of face images and transfer all of them into a high dimensional space by extracting their feature vectors. Then, key-frames of different identities are decided by clustering algorithms such as K-means [9, 14]. The optical approach extracts key-frames according to inter-frame human body's motion extracted by optical flow algorithm such as Lucas-Kanade [19].

For quality based approach, Nasrollahi et al. used four metrics: symmetry degree of face image, sharpness, contrast and brightness value to generate a face image's quality value and use it as the criteria to select key frames [15]. Similarly, Anantharajah et al. also applied face image metric based quality value generation system for face image clustering in news videos [3]. One obvious advantage of face quality metric based approach is its low computation overhead. For instance, Qi et al. achieved a faster than real-time (≥30fps) processing speed on HD videos with 1920 × 1080 resolution by applying quality metrics based approach [18]. But for the purpose of generating a final face quality value, the previous researches [15, 3, 18] associate pre-defined empirical weights with different face image quality metrics. Applying these fixed weights is not capable nor adaptive in dealing with different videos with various background, lighting condition, head poses, different FR methods and other conditions. To fix this problem, Chen et al. [6] proposed a learning-to-rank based method to elevate face quality assessment's capability and adaptiveness to videos with various conditions and different face recognition method. During the feature vector weight training stage, they applied three categories of face image databases: high quality face images took under controlled environment, face images captured from uncontrolled environment, and non-face images. Then, all different types feature vectors: HoG, Garbor, Gist, LBP and CNN are applied with different trained weights and combined to one rank-based quality score (RQS) vector. At last, the RQS vector is transformed into a face quality score between 0 and 100 by a linear function. Moreover, there is another variant of learning to rank approach proposed in [13]. In this work, the mismatch between training and testing images is considered as another impact factor and the visual quality of face image. As a result, the overall accuracy is further improved compared with the original work of Chen et al. [6]. Vignesh et al. proposed another learning based approach which utilizes a convolutional neural network (CNN) based face quality assessment module for frame selection of video with faces [25]. With the enhanced feature extraction and learning capabilities brought by deep neural network (DNN), their frame selection scheme outperformed the rank-based approach [6] without extracting pre-defined face features and learned different weights for different feature vectors in their experiment. But the FR engines in their work are all feature based, still lacking of deep learning based FR frameworks. What's more, evaluating only on Top-4,8,16 ranks accuracy in this paper [25] is not comprehensive enough to evaluate a FR system and the performance improvement brought by frame selection engine. What's more, the relatively simple CNN architecture in their FQA module can be further improved and evaluated.

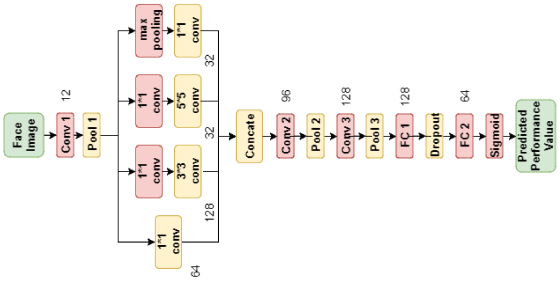


Figure 3: CNN architecture in FQA module

Hence, in this work firstly we introduce a more advanced, brand new CNN architecture for face quality assessment and evaluate the performance of our CNN based FQA engine. Secondly, our work follows the deep neural network based FR framework described in Section 1.1 and Section 1.2. Thirdly, we employed Receiver Operating Characteristic (ROC) curve, Detection Error Tradeoff (DET) curve and Cumulative Match Curve (CMC) as performance metrics to evaluate our proposed scheme on PaSC, ChokePoint, FiA datasets and HD videos took by ourselves. We deem these performance metrics can reveal more of performance trade-off on true identification, false positive and false negative identification, which makes evaluations more complete. Lastly, we also dive deeper into practical applications with GPU acceleration and evaluate the processing speed and video data volume reduction ratio.

3. CNN Based Key Frame Extraction

3.1. CNN structure

As shown in Figure 3, our CNN network has three convolution layers, three following pooling layers, two full connection layers and final sigmoid output node. In addition, we applied an inception module at the early stage of our CNN between first and second convolution layers. The key idea behind the inception module is to concatenate different features extracted by convolution kernels with different sizes. Hence, with the consideration of extracting both fine grain and coarse grain features of face images in this work, we applied the inception module here with four paths: 1×1, 3×3, 5×5 convolution and 3×3 max pooling. The other reasons for this implementation are: first, we did experiments with more than one inception module and found no significant performance improvement. Hence, to reduce the computation overhead, we implemented only one here. Second, implementing the inception module at early stage can extract more face features or loss less details than implementing after second or third convolution layer. The other layer configurations of our CNN network is also shown in Figure 3. The numbers next to each layer are the number

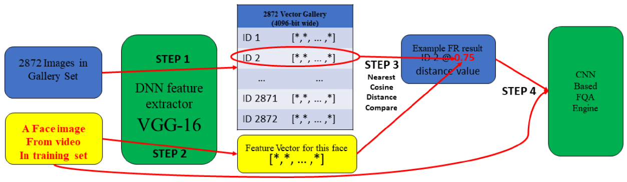

2872 Images in Gallery Set
A Face image From video In training set
STEP 1
DNN feature extractor
VGG-16
STEP 2
2872 Vector Gallery (4096-bit wide)
ID 1 [*,*, ... ,*]
ID 2 [*,*, ... ,*]
... ...
ID 2871 [*,*, ... ,*]
ID 2872 [*,*, ... ,*]
Feature Vector for this face [*,*, ... ,*]
STEP 3
Nearest Cosine Distance Compare
Example FR result ID 2 @ 0.75 distance value
STEP 4
CNN Based FQA Engine

Figure 4: CNN training procedure for PaSC dataset

of convolution kernels or number of nodes in full connection layers. Besides, face images are all scaled to 64 × 64 as input and all pooling operations are 3 × 3 max-pooling. The single node with sigmoid function is used to generate numeric output of predicted FR performance value, which has a range of 0.0 to 1.0.

3.2. CNN Training

On a theoretical aspect, let a training sample set S = { s_1, s_2, ..., s_{N-1}, s_N } be a set of N face images. For each sample s_i, we obtain a true face recognition performance value $P_{FR}(s_i)$ generated by FR back-end and a predicted performance value $P_{FQA}(s_i)$ generated by our CNN-based FQA module. Specifically, the $P_{FR}(s_i)$ value is the cosine similarity value between sample's feature vector and its nearest vector in gallery set, which is describe in Section 1. The loss function of training set $L(S)$ is defined as the euclidean loss which shown in Equation 2. Hence, the training target is to minimize the total loss $L(S)$ of the training set. In our experiment, the Nesterov accelerated gradient (NAG) [16] optimizer is applied to achieve the training target.

$$L(S) = \frac{1}{N}\sum_{n=1}^{N}||P_{FR}(s_i) - P_{FQA}(s_i)||^2 \qquad (2)$$

On implementation aspect, we use a specific example shown in Figure 4 to illustrate the CNN training procedure. The training with PaSC dataset follows four steps: 1) There are 2872 still images in PaSC's training set used as gallery. We forward all these still gallery images to DNN feature extractor and generate the feature vector gallery. 2) There are 280 videos in PaSC's training set. We detect faces from these videos and generate feature vectors by forwarding all faces to DNN feature extractor. 3) We generate faces FR performance value by looking up the vector gallery, and find the one with the nearest cosine distance value as the FR result. For instance, as in Figure 4, if we find ID2 in vector gallery is the nearest one to the training sample with a cosine similarity of 0.75, we use 0.75 as the FR performance value. In other word, the FR performance value stands for query sample's highest acceptability by the gallery set. 4) WE feed all face images and their corresponding FR values to train our CNN based FQA engine. For other datasets we tested in this work, we follow the same FQA CNN training framework.

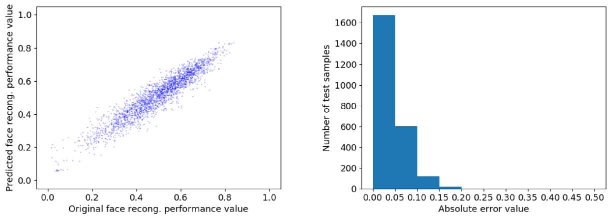
3.3. Performance of our CNN model

We conducted a performance analysis by using 10% of face image samples extracted from the portal 1 (P1) of 1-person videos in ChokePoint dataset, which has 2,417 face image samples in total. For all test samples, firstly, we perform FR on them to generate true FR performance value. Then, we forward them through our CNN FQA model to predict their FR performance value. Finally, the performance of our CNN based FQA model can be evaluated by the correlation and the absolute error between samples' true FR value and predicted FR value. **[R3A4] To emphasize, We did split the data when conducting the experiment, where the testing data for correlation verification is not appeared in training stage.** Figure 8 shows the performance analysis of our CNN model. In sub-figure 5a, the x-axis is the true FR performance value of test sample and the y-axis is the predicted FR performance value output by our CNN model. Overall, our CNN FQA module achieves a correlation of 94.30% , which means a strong linear relation between sample's true performance value and predicted performance value. In sub-figure 5b, we also show the absolute error histogram between test sample's true FR performance value and predicted FR performance value. From this figure we can see that 94.08% of samples have the absolute prediction error less than 0.1 and only 5.83% samples have a larger error between 0.1 and 0.2. To conclude, Figures 5a and 5b prove that our CNN model is highly effective to perform as a FQA module in KFE engine for face in video recognition.

4. Evaluations

In order to verify the performance improvements brought by our KFE engine on FR back-end, we arrange our experiments into two categories:

Theoretical performance evaluation category with abundant 1-person videos from PaSC and ChokePoint datasets. With the consideration of meeting the conceptual idea of KFE - choosing the best frame for one specific identity, we let all frame selection methods under evaluation to choose one best frame and compare with all-frame based FR approach for every 1-person video/clip. This set-



(a) Correlation between true and predicted FR performance value

(b) Absolute error histogram

Figure 5: Performance of Our CNN model

ting is valid as in this category of videos there is no need of applying face tracking module, since we have the prior knowledge of the identify of “1-person”. Moreover, there are lots of face tracking approaches and implementations, which will make the evaluation less fair since the number of selected key-frames can be affected by the performance of the face tracking module, for example, lost detection or re-detect the same person with new label. For evaluation metrics, ROC curve and DET curve are used as evaluation metrics to reveal the precision trade-off of true Positive Rate (TPR), False Positive Rate (FPR) and False Negative Rate (FNR).

Practical performance evaluation category with multiple people videos. [R2A1] Compared with 1-person videos in existing datasets, we also evaluate our KFE engine under the crowd case to a certain extent, such as the two long videos from ChokePoint dataset with 29 identities and other two HD videos we took ourselves, to prove our KFE engine is capable of handling multiple-people case. Compared with abundant 1-person videos, the scarcity of multiple people videos in existing datasets makes ROC and DET curves less confident, because we can only generate limited face sample. In this case, the CMC curve is a suitable metric to reveal system’s recognition accuracy without affecting test sample number. On the other hand, face tracking is unavoidable in multi-people videos. Hence, more than one key-frames can be extracted for the same person. We will evaluate the data reduction ratio in this evaluation category to prove the effectiveness of our KFE engine. Besides, we also evaluate HD videos (1920 × 1080) to verify the processing ability of our GPU accelerated KFE engine. **[R3A1] We implemented two different FR back-ends trained with different face databases because the performance of our key-frame extraction (KFE) engine will be impacted by the back-end face recognition (FR) engine. Moreover, we also want to validate the KFE engine design we proposed as a general approach in selecting frame(s) with the best face quality and it can work with different FR back-end.**

4.1. Evaluation Datasets

PaSC dataset [4]: The Point and Shoot Face Recognition Challenge (PaSC) dataset has 9,376 still images and 2,802 1-person videos of 293 different identities. **[R3A5] The PaSC dataset also provides several file lists to describe which files can be used for training or testing, which files are still images that can be used for building the gallery. Hence, we followed the official ground-truth files to build our gallery in experiment.** For evaluation, we use 4,688 still images (16 still images/person on average) to form gallery set and all 2,802 videos as the query set. There is also a small training set provided, which consists of 2,872 still images and 280 videos.

ChokePoint dataset [27] : The ChokePoint dataset contains 16 footages with people passing one by one, and 2 footages with crowds. In our experiments, the 90% faces extracted from Portal 1 (P1) of 1-person videos are used for training our FQA CNN. Since there are no official gallery set provided in ChokePoint dataset, we follow the same configuration as PaSC with 16 images/identity to form the gallery set. Hence, for 29 identities in this dataset, we picked 464 images in total from the remaining 10% data of P1 to form the gallery set.

FiA dataset [8] : The Face in Action (FiA) dataset contains three sessions of videos taken at different date and each session consists of two parts: indoor and outdoor. We design a more challenging experimental scenario where we build the gallery set from the indoor videos in Session 1, and use 233 identities and randomly pick 7 gallery images for each identity. We train our CNN based FQA by using the outdoor videos in Session 1 and use the outdoor videos in Sessions 2 and 3 as test set. As we can see, the gallery building is to imitate people taking registration photos in indoor controlled environment and the testing is to imitate the real application in outdoor uncontrolled environment.

Full HD videos: We also took two HD videos (1920 × 1080) for testing processing speed and data volume reduction. The first video shown in Figure 6a was taken in a corridor with 4 different people, and the second video shown in Figure 6b was taken in the hallway with 22 identities.

(a) Video took in corridor (b) Video took in hallway

Figure 6: Scenes of HD videos

4.2. Experimental Settings

Settings for DNN in FR back-end: The first DNN feature extractor we use in FR back-end is VGG-16 network [17] trained with VGG face dataset. We directly applied the pre-trained Caffe model by VGG team [17]. We use the second-to-last fully-connected layer as the deep feature vector of face image, and the vector width is 4096 float point numbers. Besides, we also used GoogLenet inception-v3 DNN architecture trained with CASIA-WebFace dataset to form another back-end. The purpose of introducing a second FR back-end is to prove our approach is compatible with different FR engine in practical cases. Hence, for the practical performance analysis in Section 4.4, we test multiple people videos with both VGG and WebFace-based FR back-ends. For web-face trained FR engine, we also use the second-to-last fully-connected layer as the deep feature vector of face image. The only difference is the width of feature vector width generated from WebFace-based is 2048.

Other methods to be compared with: In our experiments for theoretical performance analysis, **[R1A1] we compared our approach against several representative schemes for video based frame selection, which use different criteria in selecting key frames**: Performing FR on all frames; Random picking; Face quality metrics based picking [18]; Face quality based pick by using Learning to rank method [6]; Reference work by using CNN based face quality assessment [25]. **[R1A1]Since the focus of this work is face-quality-assessment-based Key-Frame Extraction (KFE) engine, we deem the organization of our experiment is fair.**

Gallery Settings: [R3A7] It is widely accepted that gallery picking can affect the FR performance, as being quantified in [12]. But we want to examine our KFE engine can work with non-ideal conditions. Hence, we did not choose the gallery in a particular way, but just randomly pick images to form the gallery, which contains both good and poor-quality images.

4.3. Theoretical Performance Evaluation Results

For PaSC dataset in Figure 7a, the ROC curve shows that our CNN-based approach outperforms all other methods and performing FR on all frames. Specifically, for global metric of AUC (Area Under Curve), our approach achieved the highest value. For the True Positive Rate (TPR) at 1% False Positive Rate (FPR), our approach is also the best. It means for reaching the same true identification rate, our approach has the least trade-off of false positive identification. What's more, from the curves' trend we can see that the advantage of our approach becomes more obvious after the 3% FPR. Figure 7d shows the DET (Detection Error Trade-off) analysis of all evaluated methods. Again, our CNN-based FQA approach outperforms all other methods with least trade-off in FNR (False Negative Rate) and FPR. Besides, our method has the lowest Equal Error Rate (EER) [1]. To conclude, the EDT analysis also proves our method has the lowest trade-off of negative effects.

For ChokePoint dataset in Figure 7b, from the ROC curve we can see that our CNN-based approach outperforms all other methods and performing FR on all frames. Specifically, for global metric AUC (Area Under Curve), our approach achieved the highest value. For the TPR at 1% and 10% FPR, our approach is also the best one. What's more, from the curves' trend we can see that the advantage of our approach becomes more obvious after the 5% FPR. Figure 7e is the DET analysis of all methods. Again, our CNN-based FQA approach outperforms all other methods with the least trade-off in FNR and FPR. Besides, our method has the lowest EER.

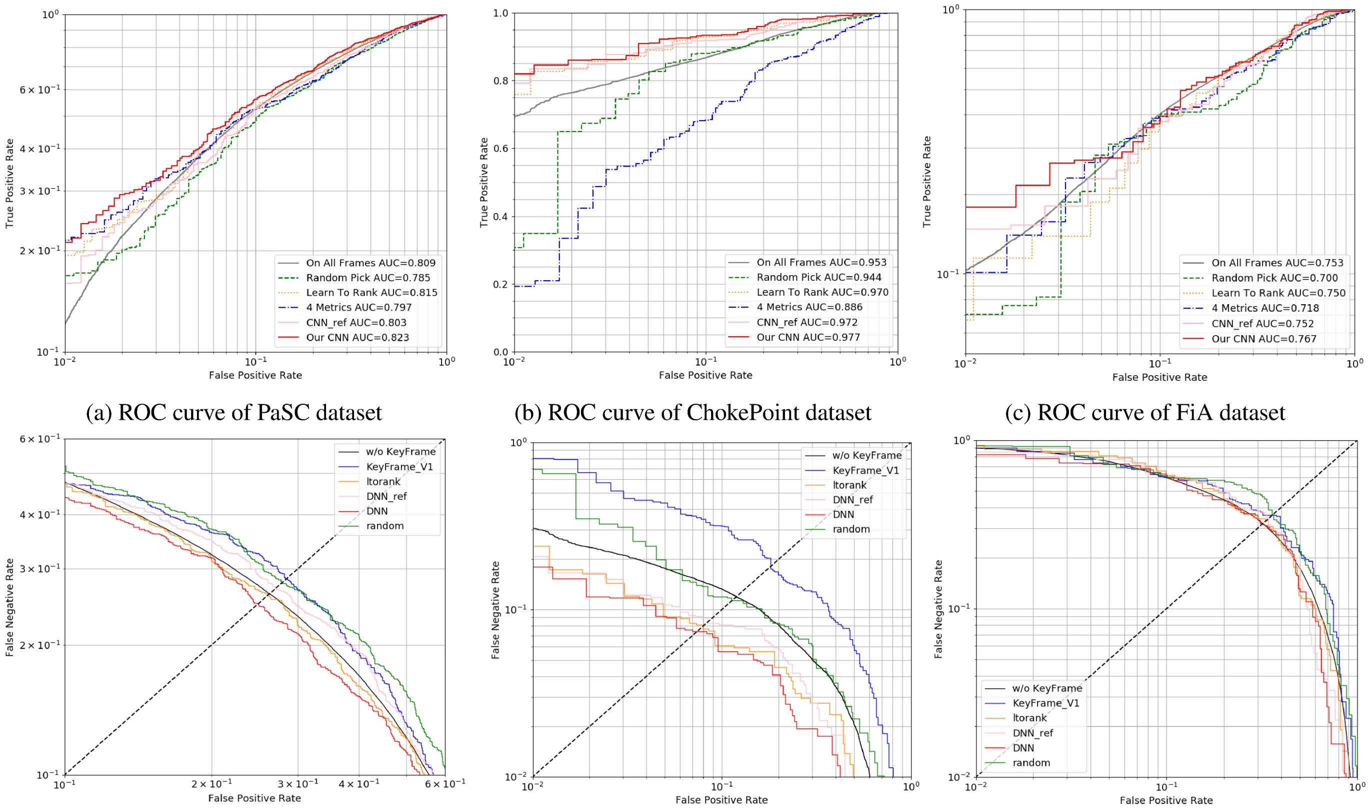
For FiA dataset in Figure 7c, our approach is the only one which could bring improvement in AUC metric compared with all frame based FR. In addition, our approach improved the TPR within a low FPR zone (1% to 5%), and works better than other methods, too. The only unsatisfiable result occurs in the zone of (0.05,0.1), which we can see that our approach performs slightly worse than all-frame based method. To analyze the reason, we went back and observed the face images with wrong FR result. We found that many of these wrong identified face images are with dull illumination, and the face images in gallery set are all with somewhat good lighting condition. Since our experiment setting is to imitate a real world process: gallery images taken in controlled environment and tested in uncontrolled environment, hence, it would be better if we can take face photos with dark and bright lighting during the registration process. On the other hand, in the DET analysis shown in Figure 7f, our method is the only one which can improve the performance of all-frame based FR. To conclude, the ROC and DET analysis with FiA dataset shows that our approach can handle more challenging scenario with time and location variation.

From the theoretical evaluation we can conclude that: first, random picking is obviously not a good approach, although it could reduce the data volume for FR; second, the face image metric based approach is better than random picking, but still has a larger performance trade-off than with all-frame based FR approach, since it is not adaptive to different conditions in videos; third, learning based approaches such as learning to rank [6], reference CNN [25] and our approach are capable of performing KFE for FR and achieve better performance than all-frame based approach, since these methods extract and learn features of faces under different conditions. Among these learning based approaches examined in the experiments, our CNN-based FQA approach has the best performance and least trade-off on FPR and FNR.

4.4. Practical Performance Evaluation Results

We evaluated the two multi-person videos in ChokePoint dataset. From the CMC in Figures 8a, 8b, 8c and 8d, we can see that our CNN based FQA method can improve the recognition accuracy compared with performing FR on all frame under both back-ends: VGG-16 network trained with VGG dataset and GoogLenet-inception_v3 trained with web-face dataset. This means in real application with multiple people, our approach can bring the benefit of improving FR accuracy as well as reducing the data volume by picking frames with good quality faces. **[R3A6] We use videos with high resolution to verify our systems processing capability. Hence, we used two HD videos with** 1920×1080 **resolution to test our system. The result in Table 2 shows that our GPU accelerated KFE engine can reach higher**

[1] EER: the cross point with 45 degree dashed line).



(a) ROC curve of PaSC dataset

(b) ROC curve of ChokePoint dataset

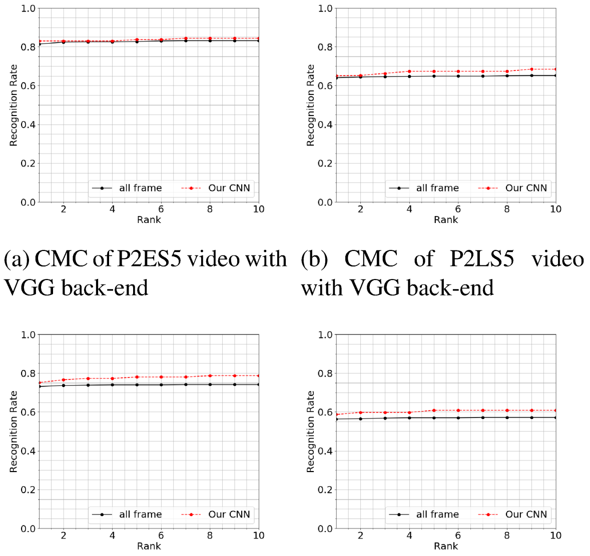
(c) ROC curve of FiA dataset

(d) DET curve of PaSC dataset

(e) DET curve of ChokePoint dataset

(f) DET curve of FiA dataset

Figure 7: Performance evaluation results on PaSC, ChokePoint and FiA datasets



(a) CMC of P2ES5 video with VGG back-end

(b) CMC of P2LS5 video with VGG back-end

(c) CMC of P2ES5 video with Web-Face back-end

(d) CMC of P2LS5 video with Web-Face back-end

Figure 8: Evaluation on crowd videos in ChokePoint

than real-time (24 or 30 fps) processing speed. Moreover, using key-frames instead of all frames for FR reduces more than 90% of frame data.

Table 2: Performance analysis on HD videos

Video	Total Frames	Processing Speed	Extracted Key-Frames	Reduction Ratio
Corridor	292	35.75fps	24	91.78%
Hallway	1508	36.92fps	134	91.11%

5. Conclusion

Face in video recognition plays a key role in video analytics and video surveillance. To address the challenges caused by variations in face quality, heavy video data volume, real-time processing demand and the need for improving identification accuracy, in this paper we propose a GPU accelerated key-frame extraction (KFE) engine with CNN based face quality assessment (FQA) module. Under the experiments of theoretical performance analysis, our KFE engine achieved the best performance among all comparing methods in ROC and DET metrics. Moreover, under the experiments of practical performance analysis, our KFE engine also shows a good capability in dealing with videos with crowds in close to real application scenarios. What's more, by applying the GPU acceleration, our KFE engine reaches higher than real-time processing speed when dealing with HD videos. For future works, we will continue to improve our KFE engine's capability to fit more uncontrolled application scenarios better. **[R2A3] What's more,**

the handling of crowd scenario is another important future direction. [R2A1] Since the focus of this work is the Key-Frame Extraction (KFE) engine, which is the stage after the face detector, we used the default face detector module of OpenCV. Hence, our work inherits the limitation of this face detector when it comes to the case of handling the crowd scenario. In addition to improve the face detector itself, we can consider digital zooming. For improving the quality of detected face image we can apply super resolution, face image artifacts mitigation and other approaches, which can be set as target of our next-stage work. Lastly, we will also consider other possible directions such as taking identity related background and object information into consideration.

Acknowledgement

This material is based upon work supported by the Center for Identification Technology Research and the National Science Foundation under Grants No. 1068055 and 1650503. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(12):2037–2041, 2006.

[2] B. Amos, B. Ludwiczuk, and M. Satyanarayanan. Openface: A general-purpose face recognition library with mobile applications. *CMU School of Computer Science*, 2016.

[3] K. Anantharajah, S. Denman, D. Tjondronegoro, S. Sridharan, C. Fookes, and X. Guo. Quality based frame selection for face clustering in news video. In *Digital Image Computing: Techniques and Applications (DICTA), 2013 International Conference on*, pages 1–8. IEEE, 2013.

[4] J. R. Beveridge, P. J. Phillips, D. S. Bolme, B. A. Draper, G. H. Givens, Y. M. Lui, M. N. Teli, H. Zhang, W. T. Scruggs, K. W. Bowyer, et al. The challenge of face recognition from digital point-and-shoot cameras. In *Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE 6th International Conference*, pages 1–8, 2013.

[5] M. Bicego, A. Lagorio, E. Grosso, and M. Tistarelli. On the use of sift features for face authentication. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*, pages 35–35. IEEE, 2006.

[6] J. Chen, Y. Deng, G. Bai, and G. Su. Face image quality assessment based on learning to rank. *IEEE Signal Processing Letters*, 22(1):90–94, 2015.

[7] T. I. Dhamecha, G. Goswami, R. Singh, and M. Vatsa. On frame selection for video face recognition. In *Advances in Face Detection and Facial Image Analysis*, pages 279–297. Springer, 2016.

[8] R. Goh, L. Liu, X. Liu, and T. Chen. The cmu face in action (fia) database. In *AMFG*, volume 5, pages 255–263. Springer, 2005.

[9] A. Hadid and M. Pietikainen. Selecting models from videos for appearance-based face recognition. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 1, pages 304–308. IEEE, 2004.

[10] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, Technical Report 07-49, University of Massachusetts, Amherst, 2007.

[11] A. K. Jain and S. Z. Li. *Handbook of face recognition*. Springer, 2011.

[12] H. Kaya, F. Gürpınar, and A. A. Salah. Video-based emotion recognition in the wild using deep transfer learning and score fusion. *Image and Vision Computing*, 2017.

[13] H.-I. Kim, S. H. Lee, and Y. M. Ro. Face image assessment learned with objective and relative face image qualities for improved face recognition. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 4027–4031. IEEE, 2015.

[14] W. Liu, Z. Li, and X. Tang. Spatio-temporal embedding for statistical face recognition from video. *Computer Vision–ECCV 2006*, pages 374–388, 2006.

[15] K. Nasrollahi and T. B. Moeslund. Face quality assessment system in video sequences. In *Biometrics and Identity Management*, pages 10–18. Springer, 2008.

[16] Y. Nesterov. Gradient methods for minimizing composite objective function. 2007.

[17] O. M. Parkhi, A. Vedaldi, A. Zisserman, et al. Deep face recognition. In *BMVC*, volume 1, page 6, 2015.

[18] X. Qi and C. Liu. Gpu-accelerated key frame analysis for face detection in video. In *2015 IEEE 7th International Conference on Cloud Computing Technology and Science(CloudCom),*, pages 600–605, 2015.

[19] U. Saeed and J.-L. Dugelay. Temporally consistent key frame selection from video for face recognition. In *Signal Processing Conference, 2010 18th European*, pages 1311–1315. IEEE, 2010.

[20] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015.

[21] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *Advances in neural information processing systems*, pages 1988–1996, 2014.

[22] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1891–1898, 2014.

[23] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[24] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[25] S. Vignesh, K. M. Priya, and S. S. Channappayya. Face image quality assessment for face selection in surveillance video using convolutional neural networks. In *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP),*, pages 577–581, 2015.

[26] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 529–534. IEEE, 2011.

[27] Y. Wong, S. Chen, S. Mau, C. Sanderson, and B. C. Lovell. Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition. In *IEEE Biometrics Workshop, Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 81–88. IEEE, June 2011.

[28] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014.

[29] C. Zhang and Z. Zhang. A survey of recent advances in face detection, 2010.