https://doi.org/10.2352/ISSN.2470-1173.2019.8.IMAWM-400

Face Set Recognition

Tongyang Liu [1]***, Xiaoyu Xiang*** [1]***, Qian Lin*** [2]***, Jan P. Allebach*** [1]***, 1. Purdue University, West Lafayette, IN, USA. 2. HP Labs, Palo Alto, CA, USA.***

Abstract

In this paper we present a Cluster Aggregation Network (CAN) for face set recognition[1]*. This network takes a set of face images, which could be either face videos or clusters with a different number of face images as its input, and then it is able to produce a compact and fixed-dimensional feature representation for the face set for the purpose of recognition. The whole network is made up of two modules, among which the first one is a face feature embedding module and the second one is the face feature aggregation module. The first module is a deep Convolutional Neural Network (CNN) which maps each of the face images to a fixed-dimensional vector. The second module is also a CNN which is trained to be able to automatically assess the quality of input face images and thus assign various weights to the images' corresponding feature vectors. Then the one aggregated feature vector representing the input set is formed inside the convex hull formed by the input single face image features. Due to the mechanism that quality assessment is invariant to the order of one image in a set and the number of images in the set, the aggregation is invariant to these factors. Our CAN is trained with standard classification loss without any other supervision information and we found that our network is automatically attracted to high quality face images, while repelling low quality images, such as blurred, blocked, and non-frontal face images. We trained our networks with CASIA and YouTube Face datasets and the experiments on IJB-C video face recognition benchmark show that our method outperforms the current state-of-the-art feature aggregation methods and our challenging baseline aggregation method.*

Introduction

The research on face set recognition has attracted more and more attention from the computer vision community [1][2][3][4][5][6]. Different from single face image recognition, which basically deal with single face feature extraction and similarity calculation, more information can be extracted from sets of faces from different identities, which are naturally composed of faces with variations in image quality, facing directions and illumination conditions. The sets are usually from face video frames and face image clusters, which are naturally not constrained with specific orders and the number of face images. Hence the key issue in face set recognition is to develop an efficient and appropriate representation of face sets, such that it can effectively gather the dominant information in a face set (e.g. information from sharper, more frontal face images) while disregarding noisy information.

One naive method for the face set recognition is to recognize the face image set as a cluster of face features that are extracted by a deep CNN [7][14], and hence in order to compare two face image sets, one needs the fused matching results from individual face feature pairs. Let n be the average number of face images in the face set, then a computational complexity of $O(n^2)$ is required per similarity comparison, which dramatically increases as the number of images goes up. This will be a critical problem when we want to build time-sensitive applications such as real time video face recognition systems. Therefore, in our discussion afterwards, we do not consider such methods as a challenging competitor to our method.

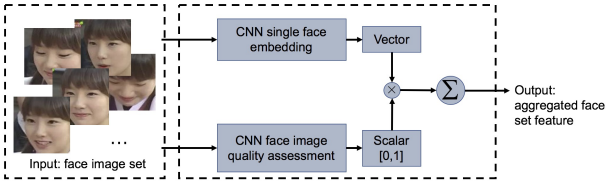


Figure 1. *The flow chart for CAN. The input face images are processed simultaneously by two parallel neural networks. The first one is a feature embedding deep CNN module which generates the features from single face images represented as fixed-dimensional vectors. And the second one is a quality assessment network that is able to automatically emphasize face features from higher quality face images while repelling the features from lower quality face images, thus forming different weighting factors. These features and their corresponding weighting scalars are then forwarded for a weighted summation such as to produce a single 1024-dimensional vector representing the input face set, which is used for recognition.*

We propose that it is desirable to develop a fixed-dimensional compact feature representation for the face sets, which is not related to the indexing and number of the images in each set. Such features should consider the information from all the images in the set while emphasizing the information from higher quality face images. Then it will allow direct, immediate computation for the set similarity or distance. One straightforward solution for generating such a representation is the strategy to take the average of the features in a set. Although state-of-the-art deep neural networks are already able to generate very efficient feature representations for different identities (meaning that taking the average of face feature is already a very competitive method for generating such representations), we believe that features generated from more frontal faces and sharper images should be preferably considered over the features generated from occluded, non-frontal and blurred face images. Hence we are looking for a smart algorithm to capture these characteristics from input images and accordingly assign different weights to their corresponding feature vectors.

Face set and video face recognition have been actively studied in the past. Many previous approaches have attempted to rep-

[1]This project is supported by HP Labs, HP Inc., Palo Alto, California, USA.

resent the face set manifolds or subspaces and compute the manifold similarity or distance for recognition [15][16]. These methods may work well in some constrained scenarios, but has limited capabilities for handling more casual and complex situations where there are large variations between image frames. Besides these manifold-based methods, there also has been prior research on aggregating features using CNNs. Yang et al. [1] proposed a method of using the attention blocks as the universal face feature quality assessment, and then aggregating them, known as a Neural Aggregation Network (NAN). Their method borrows the differentiable memory addressing mechanisms from the Neural Turing Machine. Although they also build a feature aggregation method based on smart weighting, the attention block that they used essentially needs to read all the feature vectors from the input before generating linear weights for them. Therefore, their proposed method needs preallocated memories and extra running time each time their method performs aggregation. In addition, the *weights* generated by their network are more like arbitrary values that are assigned to each of the feature vectors depending on the context of the input. Thus the relationship between these generated values and the *quality* of the original images is weak. Therefore the network cannot be treated as a universal face image feature quality assessment. To compensate for this shortcoming, Liu et al. [2] proposed a feature aggregation network called Quality Aware Network (QAN) which uses a Fully Convolutional Network (FCN) [17] to simultaneously generate face feature representation and assign weights to them. Even though this method emphasizes more the relationship between the generated feature weights and face image quality, the combination of feature generation networks essentially makes their aggregation network inflated. In practical scenarios where thousands of competitive CNN face feature extractors are available, this method shows its limit by not taking advantage of these rapidly advancing approaches for single face embedding.

In order to overcome the issues from these previous methods, we propose a novel feature aggregation method that not only utilizes the state of the art face feature extractor, but also serves as a universal face image quality measurement. The proposed method for face set recognition method contains two methods, as shown in Figure 1. Each module is a CNN which is trained separately. The first one is a face feature extractor using a deep CNN. The second one is the aggregation module which incorporates a quality assessment neural network that serves as weights generator for the features obtained from the first module. As the key component in our feature aggregation module, it will be discussed in detail in the following sections.

In summary, we want to look for an efficient and smart adaptive weighting scheme to linearly combine single face features from a face set together to form a discriminative face set representation. We designed a neural network to adaptively weigh features depending on the assessment of their original face images. We name our network the Cluster Aggregation Network (CAN), for which the parameters are trained through supervised learning using only the information of a normal face recognition task, e.g. face identities without any other extra supervising signals.

Face feature embedding module

The face feature image embedding module of our method is a deep CNN, which embeds each image from a face set to a feature vector representation. In order to leverage modern CNNs with state of the art performance, we adopt the recent proposed SphereFace feature extractor [3], which produces a 1024-dimension feature vector for each of the input face images. In the rest of the paper, we will refer to the employed SphereFace network as "the CNN". The training of the CNN used a standard face classification and verification process and we trained it on the publicly available CASIA WebFace dataset [4]. For evaluating the efficiency of our trained CNN, we tested its accuracy on 10 different folds of the LFW dataset [5], as shown in table 1. We can see that SphereFace CNN model is accurate in identifying single faces; and thus we can use it as a robust and efficient single face embedding.

Table 1. Evaluation result for CNN feature extractor on the LFW dataset

Fold	Accuracy
1	99.17%
2	99.00%
3	99.00%
4	99.50%
5	99.00%
6	99.17%
7	99.17%
8	99.00%
9	99.83%
10	99.33%

Face feature aggregation module

In this section we discuss the feature aggregation module which essentially takes the feature vectors from the feature embedding module, and then generate face set representation. Consider that we are recognizing n pairs of face image sets, each containing a varying number of images K_i. Therefore, each face image set X^i is represented as $X^i = \{x_1^i, x_2^i, ..., x_k^i\}$, where $x_k^i, k = 1, 2, ..., K_i$ is the k-th image in the face set. Each image has a corresponding feature representation ϕ_k^i, which is extracted by the feature embedding module. For better readability, the superscript i is omitted in the following text. By forwarding the images $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_k\}$ to the quality assessment network, we will have a quality score normalized to a value between zero and one $\{\sigma_1, \sigma_2, ..., \sigma_k\}$, which corresponds to the set of the input images. We then set $\sigma_j, j = 1, ..., k$ as the *weights* for the feature vectors $\phi_j, j = 1, ..., k$. Hence the aggregated feature $\mathbf{r}$ is represented as

$$\mathbf{r} = \frac{\sum_{j=1}^{k} \sigma_j \phi_j}{\sum_{j=1}^{k} \sigma_j}. \quad (1)$$

In this way, we can see that the aggregated feature vector is a weighted summation of the input single face feature vectors. Therefore, the output feature vector is of the same dimension as a single face feature vector; and it is independent of the order or the number of input face images in the face set.

Note that here we are directly assigning the quality scores as the weights for the feature vectors. We observed that the images that are preferred by our quality assessment network are more

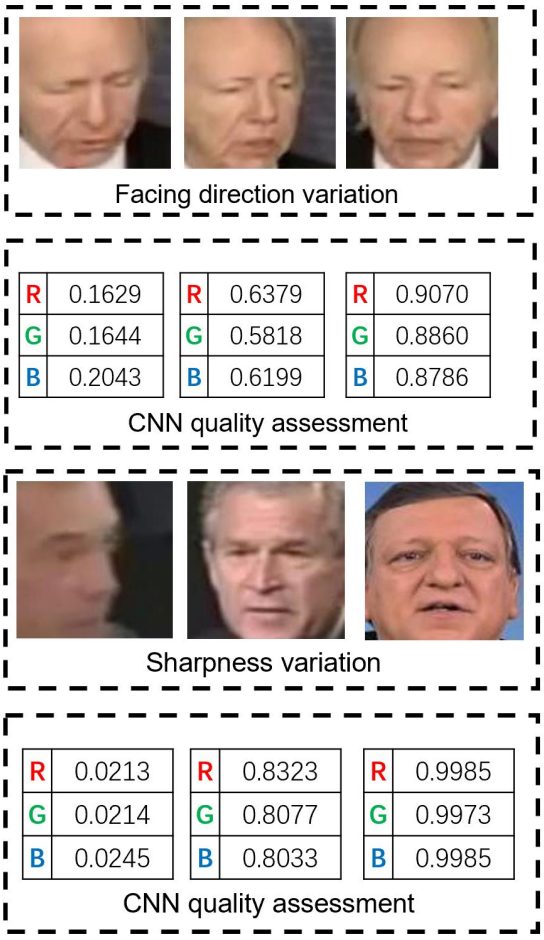


Figure 2. *Quality score generated on the three channels of face images by our quality assessment network. The top three images show the variation of face poses. From left to right: face side and down, face side, frontal face. The bottom three images show the variation of image sharpness. From left to right: most blurred face image, medium blurred face image, clear face image.*

frontal and sharper, thus the network tends to assign higher quality scores to those images. In the mean time, blurred and non-frontal face images are repelled, leading to the result that the network tends to assign lower quality scores to them. An example of the face image quality assessment by our network is shown in Figure 2, from which we are able to see that the scores are ascending from left to right, matching with the fact the face images from left to right are becoming more frontal and sharper. We presume that sharp and frontal face images are much more critical in the decision of recognition than blurred and side faces. Therefore, Equation 1 will make the face set recognition system focus on good images, while ignoring bad ones.

Quality assessment neural network

Our quality assessment module for face images is actually a Convolutional Neural Network, as shown in Figure 3. This network has a simple structure, and the process of generating quality score is a one-time network forwarding. Examples of image quality scores output from our network are shown in Figure 2. The

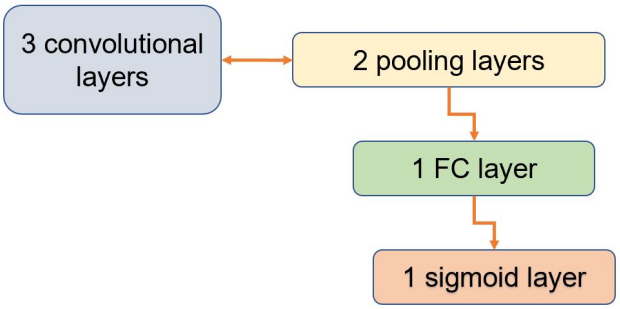


Figure 3. *Network structure of our quality assessment CNN. The input images are forwarded to 3 convolutional layers and 2 pooling layers, followed by a Fully Connected (FC) layer, and the quality scores are obtained through the sigmoid layer.*

network takes RGB color images with size $224 \times 224 \times 3$ and the sigmoid layer's output has size 1×3, which corresponds to the Red, Green and Blue channels from the input images. Different from NAN which depends on input context to generate feature weights, our quality score is only related to one input image and the parameters of the neurons in the network, which are trained with standard face recognition techniques without any other supervision signals, such as how good or bad the image is.

In order to effectively train the network, we compose our training data $\mathbf{y}$ of three parts: $\mathbf{y}_a$, $\mathbf{y}_b$ and $\mathbf{y}_c$. Each of them contributes to one third of the training samples. $\mathbf{y}_a$, $\mathbf{y}_b$ are the sets of faces that come from the same person while $\mathbf{y}_c$ comes from a different person. By forwarding $\mathbf{y}$ throughout the networks, we get the output feature representations for the three sets, noted as $\mathbf{r}_a$, $\mathbf{r}_b$ and $\mathbf{r}_c$. We want the distance between $\mathbf{r}_a$ and $\mathbf{r}_b$ be as small as possible while the distance between $\mathbf{r}_a$ and $\mathbf{r}_c$ be as large as possible. Hence we let the back propagation process to minimize the combination of the following two loss functions:

$$l_1 = \lambda \|\mathbf{r}_a - \mathbf{r}_b\|_2 + (1-\lambda)\max(0, m - \|\mathbf{r}_a - \mathbf{r}_b\|_2) \qquad (2)$$

and

$$l_2 = \|\mathbf{r}_a - \mathbf{r}_b\|_2 - \|\mathbf{r}_a - \mathbf{r}_c\|_2 + \delta, \qquad (3)$$

where the norms indicate Euclidean distance between vectors.

We see that since $\mathbf{r}_a$, $\mathbf{r}_b$ represents the same identity while $\mathbf{r}_c$ represents different identities, therefore minimizing the loss as described in Equation 5 will make sure that $\mathbf{r}_a$ and $\mathbf{r}_b$ is the closest, thus having highest similarity. Meanwhile, we note that minimizing Equation 3 is equivalent to minimizing the distance between $\mathbf{r}_a$, $\mathbf{r}_b$ while maximizing the distance between $\mathbf{r}_a$, $\mathbf{r}_c$, since δ is a constant value. Therefore, in the training process, the loss functions regulate the parameters of the neurons such that they can gradually learn to ignore the low quality images in the face set that prevent the distance between $\mathbf{r}_a$ and $\mathbf{r}_b$ from decreasing and $\mathbf{r}_a$ and $\mathbf{r}_c$ from increasing. And finally the neurons are trained to automatically protrude higher quality face images while repelling hard samples from face sets.

Actually, Equation 3 is known as *triplet loss* [7] and Equation 2 is called *contrastive loss* [6], in which $\lambda = 1$ if the pair (a,b) comes from the same identity and $\lambda = 0$ otherwise. Therefore, in

our case, Equation 2 can be further simplified as:

$$l_2 = \|\mathbf{r}_a - \mathbf{r}_b\|_2. \tag{4}$$

And the combined loss function l is:

$$l = l_1 + l_2 \tag{5}$$

In the training process, the standard backpropagation will adapt the network parameters such as to minimize the average loss as described in Equation 3 and 5.

Training details

As mentioned before, the single face embedding CNN and quality assessment CNN are trained separately in our work. To train the single face embedding CNN, we use about 45K images in the CASIA WebFace dataset [4] to perform single image-based recognition. After training, the CNN feature extractor is fixed and we focus on training the aggregation module.

To train the quality assessment CNN, the way of preparing data is that we firstly detected, cropped and aligned faces from the YouTube Face dataset. After doing this, all the faces were at the same horizontal and vertical level. Then we used the single face embedding CNN to generate corresponding feature vectors from the aligned face images. After this process we prepared around 13K image-feature pairs. Then we trained on the prepared dataset with standard back propagation and a SGD solver [8]. We set the learning rate be 0.001 and batch size be 24. We trained on an NVIDIA^{TM} 1080Ti GPU with CUDA [18] enabled; and it took around 4 hours to finish the training process.

Baseline method

Since our goal is to develop a smart feature aggregation method, we want to compare the results with simple aggregation approaches such as average weighing to see whether the smart weighing really helps to improve the face set recognition accuracy. Similar to Equation 1, the baseline method can be described as:

$$\mathbf{r} = \frac{\sum_{i=1}^{k} \mathbf{f}_i}{k}, \tag{6}$$

where $\mathbf{f}_i$ is the same CNN feature used by CAN. We see that the baseline method also generates a feature vector with the same dimension as a single face feature.

Results on IJB-C dataset

The IJB-C dataset contains face images and videos that are captured from situations in the wild. It features a wide variety in pose, illumination and other kinds of imaging conditions, thus it is very challenging. We tested on the video frames from IJB-C, which has 500 identities with 2042 videos in total and around 11 frames for each person's video. We compared our face set recognition accuracy with reported results on IJB-C's 'compare' protocol for 1:1 *face verification* from current state of the art methods and also our own baseline method. And it shows that our methods compete over current state of the art and our own baseline method, as shown in Table 2.

We see from the results that QAN outperforms its previous state of the art by 0.65% and our proposed CAN outperforms QAN by 0.8%. In addition, it is noteworthy that CAN outperforms our baseline method by 2.32%, meaning that our smart aggregation actually works better than the naive aggregation method. An example of the failure case from using the baseline method, while non-failure from CAN is shown in Figure 4. We see that for face set on the left, the image on the right is much more blurred than the image on the left, in addition, the pose of the face on the right is not as frontal as the one on the left. Therefore, by forwarding the two face images to CAN for quality assessment, our network is able to smartly generate a higher score for the image on the left, while much lower score for the image on the right. As for the face set on the right, the images actually do not vary a lot with sharpness but vary in face pose. Hence our proposed CAN is also able to detect these variations and adaptively assign weights to the images. We notice that for this image set, from left to right the facing angle of the person is gradually decreasing, thus interestingly, weights generated by CAN are also in descending order. This is actually consistent with human cognition.

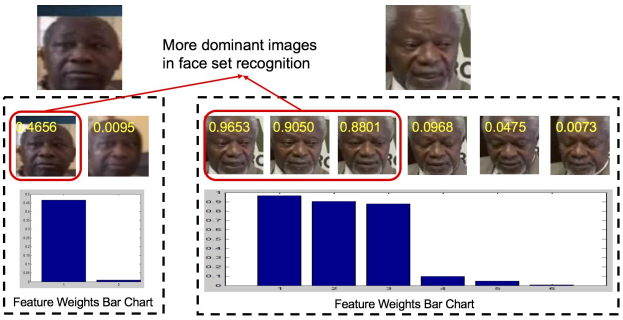


Figure 4. *An example of successful verification for two face sets using CAN but unsuccessful verification using the baseline. The face set on the left and on the right are recognized as different persons by the CAN, but are mistakenly recognized as the same person by the baseline. The reason for the difference in recognition comes from the weighting scheme of the baseline and the CAN. The yellow digits on top are the weights of face features generated by CAN, which have been averaged over three channels. The Bar Chart visualizes the variation of the feature weights.*

Table 2. Verification Accuracy comparison of state of the art methods, our baseline methods and our proposed CAN network on IJB-C dataset

Method	Accuracy (%)
EigenPEP [19]	84.8
DeepFace-single [14]	91.4
DeepID2+ [20]	93.2
CNN+Baseline	94.65
FaceNet [7]	95.12
NAN [1]	95.52
QAN [2]	96.17
CAN(ours)	**96.97**

Conclusions

We have presented a Cluster Aggregation Network for face set representation and recognition. It gathers all the input frames and uses an adaptive weighing schematics based on the smart

assessment for face images quality to generate a set of variable weights for the input, resulting a compact representation of the face image set. This method is simple, competitive, and can also be used in many scenarios, such as video face recognition, face clustering, and many other vision tasks.

References

[1] J. Yang, P. Ren, D. Zhang, D. Chen, F. Wen, H. Li and G. Hua. "Neural Aggregation Network for Video Face Recognition." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5216-5225, 2017.

[2] Y. Liu, J. Yan and W. Ouyang. "Quality Aware Network for Set to Set Recognition." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4694-4703, 2017.

[3] Z. Cui, W. Li, D. Xu, S. Shan, and X. Chen. "Fusing Robust Face Region Descriptors via Multiple Metric Learning for Face Recognition in the Wild." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3554-3561, 2013.

[4] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang. "Probabilistic Elastic Matching for Pose Variant Face Verification." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3499-3506, 2013.

[5] H.Mendez-Vazquez, Y.Martinez-Diaz, and Z.Chai. "Volume Structured Ordinal Features with Background Similarity Measure for Video Face Recognition." International Conference on Biometrics (ICB), 2013.

[6] L. Wolf and N. Levy. "The SVM-minus Similarity Score for Video Face Recognition." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3523-3530, 2013.

[7] F. Schroff, D. Kalenichenko, and J. Philbin. "FaceNet: A Unified Embedding for Face Recognition and Clustering." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 815-823, 2015.

[8] L. Bottou. "Stochastic Gradient Descent Tricks. Neural Networks: Tricks of the Trade." Springer, 2012.

[9] L. Wolf, T. Hassner and I. Maoz. "Face Recognition in Unconstrained Videos with Matched Background Similarity." IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 529-534, 2011.

[10] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj and L. Song. "SphereFace: Deep Hypersphere Embedding for Face Recognition," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6738-6746, 2017.

[11] D. Yi, Z. Lei, S. Liao, and S. Z. Li. "Learning Face Representation from Scratch." arXiv preprint arXiv:1411.7923, 2014.

[12] 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, University of Massachusetts, Amherst, pp. 07-49, 2007.

[13] R. Hadsell, S. Chopra, and Y. LeCun. "Dimensionality Reduction by Learning an Invariant Mapping." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), volume 2, pp. 1735-1742, 2006.

[14] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. "DeepFace: Closing the Gap to Human-Level Performance in Face Verification." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1701-1708, 2014.

[15] R.Wang, S.Shan, X.Chen, and W.Gao. "Manifold-Manifold Distance with Application to Face Recognition based on Image Set." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1-8, 2008.

[16] P. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa. "Statistical Computations on Grassmann and Stiefel Manifolds for Image and Video-Based Recognition." IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), pp. 2273-2286, 2011.

[17] E. Shelhamer, J. Long, and T. Darrell. "Fully Convolutional Networks for Semantic Segmentation." IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), pp. 640-651, 2017.

[18] J. Nickolls, I. Buck, M. Garland, and K. Skadron. "Scalable Parallel Programming with CUDA." pp. 40-53, 2008.

[19] H. Li, G. Hua, X. Shen, Z. Lin, and J. Brandt. "Eigen-PEP for Video Face Recognition." Asian Conference on Computer Vision (ACCV), pp. 17-33, 2014.

[20] Y. Sun, X. Wang, and X. Tang. "Deeply learned face representations are sparse, selective, and robust." IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2892-2900, 2015.

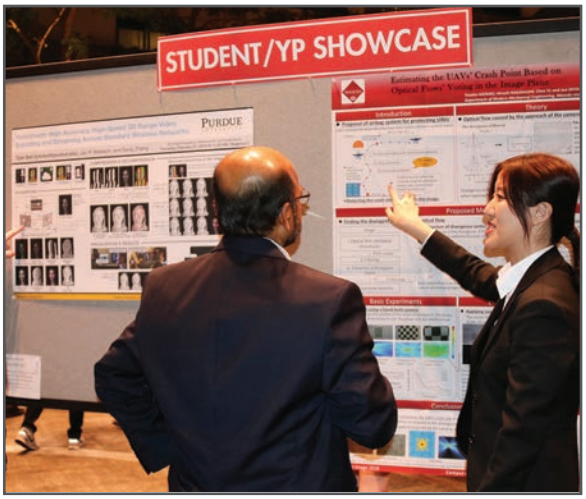
JOIN US AT THE NEXT EI!

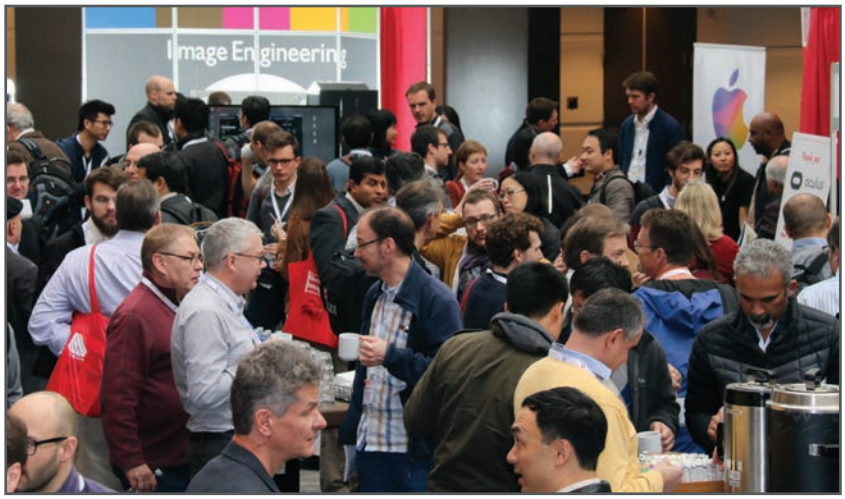
IS&T International Symposium on

Electronic Imaging

SCIENCE AND TECHNOLOGY

Imaging across applications . . . Where industry and academia meet!





• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

IS&T
imaging.org