Speeding-up Age Estimation in Intelligent Demographics System via Network Optimization

Zhenzhen Hu[*†], Peng Sun[†], Yonggang Wen[†]

[*]School of Computer and Information, Hefei University of Technology, Hefei, China
[†]School of Computer Science and Engineering, Nanyang Technological University, Singapore

huzhen.ice@gmail.com, {sunp0003, ygwen}@ntu.edu.sg

Abstract—Age estimation is a difficult task which requires the automatic detection and interpretation of facial features. Recently, Convolutional Neural Networks (CNNs) have made remarkable improvement on learning age patterns from benchmark datasets. However, for a face "in the wild" (from a video frame or Internet), the existing algorithms are not as accurate as for a frontal and neutral face. In addition, with the increasing number of in-the-wild aging data, the computation speed of existing deep learning platforms becomes another crucial issue. In this paper, we propose a high-efficient age estimation system with joint optimization of age estimation algorithm and deep learning system. Cooperated with the city surveillance network, this system can provide age group analysis for intelligent demographics. First, we build a three-tier fog computing architecture including an edge, a fog and a cloud layer, which directly processes age estimation from raw videos. Second, we optimize the age estimation algorithm based on CNNs with label distribution and K-L divergence distance embedded in the fog layer and evaluate the model on the latest wild aging dataset. Experimental results demonstrate that: 1. our system collects the demographics data dynamically at far-distance without contact, and makes the city population analysis automatically; and 2. the age model training has been speed-up without losing training progress or model quality. To our best knowledge, this is the first intelligent demographics system which has potential applications in improving the efficiency of smart cities and urban living.

Index Terms—Intelligent demographics, age estimation, deep learning, parallel computing

I. Introduction

Age estimation aims to automatically predict the exact age or age group of a facial image based on the visual features. Different from other kinds of facial information such as identity and gender, human aging is generally a slow and complicated process which makes the accurate prediction of a given facial image a challenging problem within the field of facial analysis. Facial age estimation has attracted much attention due to its potential applications in video surveillance, demographic statistics collection and business intelligence. Recently, deep learning schemes, especially Convolutional Neural Networks (CNNs), have been successfully employed for many tasks related to facial analysis including face detection, face alignment [1], face verification [2], and demographic estimation [3]. For facial age estimation, CNNs have been applied to learning aging features directly from large-scale aging dataset [4] and the deeply-learned aging patterns

lead to significant performance improvement on benchmark datasets [5], [6].

Although a number of deep-learning based algorithms have been successfully developed for facial age estimation, we still face the challenges for real-world applications. Specifically, the challenges are from application-level to system-level:

1) *Accuracy.* The performance of age estimation is not as accurate as other kinds of demographic information such as identity and gender. How to improve the estimation performance remains a challenge research problem in computer vision.
2) *Latency.* Most of the age estimation models are tested on the benchmark datasets which are mainly front-view and neutral and only have a single face. While in the real-world surveillance video frames, there are often several faces appearing simultaneously. The age estimation will be time-consuming when the number of people in an image/frame is increasing.
3) *Model Training.* Existing distributed deep learning systems, such as Caffe [7] and TensorFlow [8] usually take a long time to learn a convergent model due to the high communication overhead [9].
4) *Online Prediction.* Intelligent demographics statistics requires monitoring timeliness. Because of the mobility of urban population, the age distribution is in a dynamic change pattern. Therefore, the age estimation result should be received and updated in time.

Aforementioned challenges motivate us to develop an efficient age estimation system for intelligent demography based on the deep learning and fog-computing techniques. This system takes advantage of surveillance videos for the smart city project, which implements the population investigation dynamically at far-distance and non-contact to make the city population analysis automatically. Given a clip of surveillance video, the age distribution analysis can be dynamic and real-time. Inspired by the previous work [4], [10], [11], the system is designed based on the three-tier fog computing architecture. When a person appears in the frame, algorithms and technologies of intelligent video analytics can extract the feature of people and explore its pattern. We optimize deep-learning based age estimation model with label distribution and K-L distance loss and evaluate the performance on the

latest aging dataset. Given sufficient demographic clues of one area, we can utilize the social statistics to analyze the dynamic demographic profile of this place.

Intelligent surveillance and demographics is a rising research topic due to the rapid development of machine learning and communication algorithms. Some research works have dedicated to solve the related problems. Yi *et al.* [12] proposed the pedestrian behaviors model to analyze the stationary crowd group influence based on the surveillance video shot. Ling *et al.* [13] utilized multi-tiered distributed infrastructure storage to analyze traffic for intelligent transportation system. Wahyono *et al.* [14] detected stationary objects in video surveillance systems via dual background model subtraction for intelligent surveillance systems. For intelligent demographics collection, Alharbi *et al.* [15] proposed a demographic group prediction mechanism from smart device users based upon the recognition of user gestures.

However, so far to our best knowledge, there is no research work to implement the facial age estimation cooperated with surveillance systems for intelligent demographics applications. Building an intelligent demographics system via city surveillance network is to improve the efficiency of services by using urban informatics and technology. The traditional demographics collection method is labor intensive and time-consuming. It can only provide the information for a certain period of time, while the demographics of a city's population is shifting over time. In this context, intelligent demographics becomes critical to the success of smart cities.

The main novelties and contributions of this work are threefold:

1) We propose a fog-computing-based intelligent demographics system to automatically collect urban population information. To our best knowledge, it is the first intelligent system for demographics.
2) We implement the state-of-the-art facial age estimation algorithm for the intelligent demography system. The experiments on benchmark datasets demonstrate the effectiveness of our algorithm
3) To improve the efficiency of the whole system, we propose a communication-efficient distributed deep learning system, which is used to train our age estimation model efficient with reduced communication overhead.

The rest of this paper is organized as follows. In Section II, we give a brief overview of the system. Then we provide a detailed description of our approach in Section III and Section IV. The experiments are reported in Section V. Finally, we draw the conclusion of this work in Section VI.

II. Intelligent Demographic System: An Overview

The framework illustration of the intelligent demography system is shown in Fig. 1. Our system is designed based on the three-tier fog computing architecture, which includes an edge, a fog and a cloud layer. Specifically, the edge layer consists of a lot of external video cameras, which are in charge of capturing image data in real-time. The fog layer is composed of fog nodes, which usually are network devices like gateway, router, switch and Access Points. These fog nodes could collaboratively share storage and computing facilities. Traditional cloud servers reside in the top-most cloud layer, and could provide sufficient storage and computing resources. In our proposed intelligent demography system, external video cameras first send their captured images to the smart gateways. These smart gateways analyze all revived images, crop detected faces, and upload these face data to the cloud servers for age estimation. The cloud servers would stream processing all received face data, and update demographic information.

A. Edge Layer

The edge layer indicates the external surveillance camera networks, which are the input source of the entire system. Since the surveillance system is spread all over the city and records different scenes and events, we focus on the people domain surveillance videos and only select the cameras set in population activity areas in this paper. As the use of video surveillance cameras grows, the video resource in one city is increasing tremendously and contains redundancy visual information. In view of the general characteristics of people activities and the computational resource, the system extracts one frame every 30 seconds and transmits it to the next layer for facial analysis.

B. Fog Layer

After the video frame capturing in the edge layer, the fog layer implements the face pre-processing, including face detection and alignment, based on the fog-computing. All the frames from the edge layer will be filtered with face detection algorithm. Face detection is carried out using the OpenCV Face Detector, which is an implementation of the Viola-Jones Face Detector [16] uses a boosted rejection cascade based on AdaBoost. Given the face location and area, the image of face region can be cropped and normalized. 68 face landmark points are located by the OMRON face alignment algorithm. Faces are aligned according to the locations of two eyes and that of the mouth. The distance of two eye centers is set to 32 pixels. In a face image, the size of face bounding boxes is 128×128. The images which contain non-frontal faces are removed.

C. Cloud Layer

This layer contains two core systems. Specifically, a distributed model training system is set up to learn a deep learning model for age estimation from the training dataset. A real-time age estimation system would leverage the learned model to process received face data, and update the demographic information accordingly.

1) Distributed Model Training with Caffe-MPI: We use Caffe [7] as the computation engine for model training. The training system aims to find optimal parameters for a deep learning model to minimize its prediction error. To achieve better performance, we leverage MPI (Message Passing Interface) to parallelize the model training in a cluster with multiple GPU nodes based on the Parameter Server (PS)
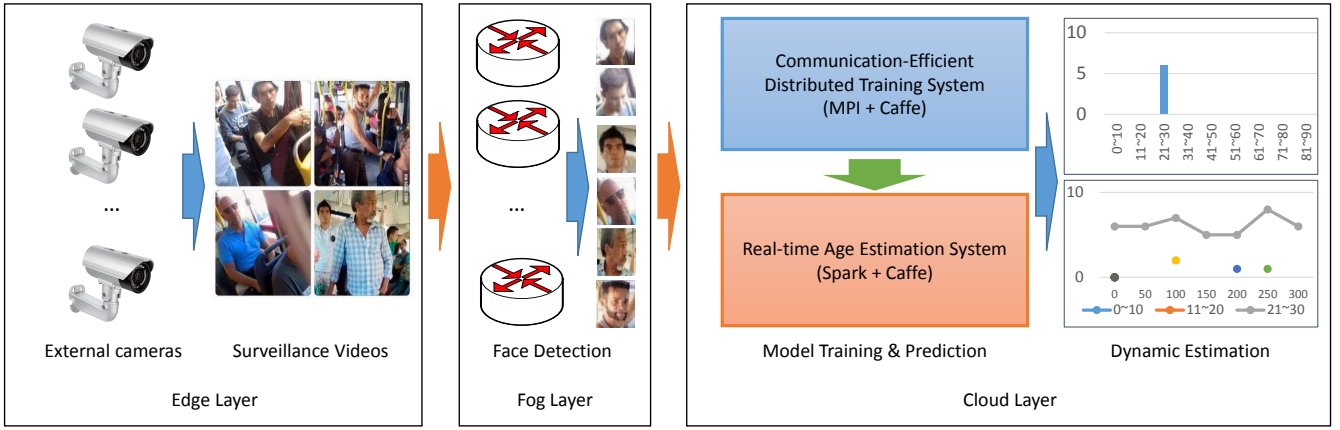
Fig. 1. An overview of intelligent demographics system. The system is a three-tier fog computing architecture, which includes an edge, a fog and a cloud layer. The edge layer captures external image data in real-time and sends them to the smart gateways. The fog layer is comprised of a set of fog nodes, like gateway, router, switch and Access Points. These smart gateways analyze all revived images, crop detected faces, and upload these face data to the cloud servers for age estimation. The cloud servers reside in top-most cloud layer, stream processing all received face data, and update demographic information.

framework [9]. When using the PS framework, Caffe-MPI evenly partitions the training dataset across multiple GPU nodes at the beginning of the training processing. During the computation, each GPU worker node fetches a batch of the assigned training data into memory, uses an optimization method like stochastic gradient decent (SGD) to compute an update value for each parameter, and pushes the updates to a server node. When receiving updates from all GPU nodes, the server node uses these data to update the model's parameters. Next, each GPU node pulls newly computed parameters from the server node for the next iteration of the computation. Caffe-MPI would perform the push-pull operations to update the model's parameters until convergence. It should be noted that each GPU node needs to push all updates and pull all parameters in each iteration, resulting in high communication overhead. In Section IV, we propose Caffe-ASU to address this problem.

2) Real-time Age Estimation with Caffe-Spark: We set up a steam processing system on the cloud layer to use the learned model to estimate received face data from the fog layer. In this system, we combine Spark Steaming [17] and Caffe. Specifically, Spark Streaming is an extension of Spark that enables scalable, high-throughput, fault-tolerant stream processing of live data streams. It structures a streaming computation as a series of stateless, deterministic batch computations on small time interval. In this system, we place the face data received every second into an interval, and run a Caffe operation on each interval to compute the age information and update the demographic information. Processed data would be stored in the local file system for future data analytics.

III. APPLICATION OPTIMIZATION: A DEEP LEARNING APPROACH FOR AGE ESTIMATION

The off-line part of the intelligent demography system is deep learning based the facial age estimation. The network architecture is shown in Fig. 2. After the pre-processing of input face images, such as face detection and face alignment, the normalized training data are feed into Convolutional Neural Networks (CNNs).

A. Basic Network Structures

Deep convolutional networks have witnessed great successes in computer vision area and many powerful network architectures have been developed, such as AlexNet [18], GoogLeNet [19], VGGNet [20], ResNet [21] and DenseNet [22]. As the surveillance videos from one city are much larger than any existing image datasets, we need to a tradeoff between the estimation performance and computational cost when building our system.

The state-of-the-art pre-trained deep network for age estimation is DEX (Deep EXpectation of apparent age) [23], an age estimation model based on the VGG-16 network architecture learned from the IMDB-WIKI dataset. This dataset includes more than 0.5 million images of celebrities from IMDb and Wikipedia, which is the largest publicly available dataset of face images with age labels.

B. Training Objective of Age Model

The original pre-trained DEX model employs the softmax function in the loss layer to train the age model. However, human aging is generally a slow and smooth process in reality, and therefore cannot be treated as a single label classification problem. In our previous work, Hu *et al.* [4] optimized the age estimation target with multi age discrete distribution vector as the ground truth label. The experimental results reflect that label distribution not only can increase the number of labeled data but also tends to learn the similarity among the neighboring ages.

In this paper, we use Gaussian distribution to model the label distribution of ages. Let $C = \{1, 2, ..., c\}$ denote the set of possible ground truth ages and $L_m = (l_m^1, l_m^2, ..., l_m^c)$ is the label distribution for the m-th image. Given a chronological age $a \in C$, the distribution of ages $\{a-2, a-1, a, a+1, a+2\}$ is calculated as $l_m^{a_i} = l_m^a \times e^{\frac{-(a-a_i)^2}{2\theta}}$, where the Gaussian function
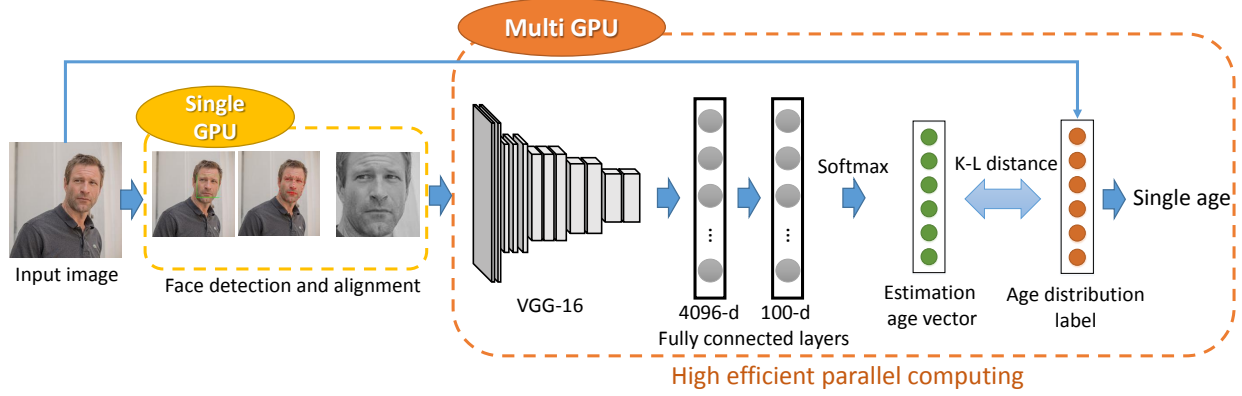
Fig. 2. The facial age estimation model learning architecture. The pre-processing of face detection and alignment is implemented on single GPU and the aligned face is the input date of CNNs. On the top layer, we use K-L distance as the loss function. The training progress is implemented on multi-GPU with parallel computing.

has the mean value a_i and variance θ. For other ages, we just let $l_m^{a_i} = 0$. Finally, a normalization process is calculated to make sure that $\sum_j^c l_m^j = 1$.

At the top layer of the deep architecture, the Kullback-Leibler (K-L) divergences distance is set to quantify the dissimilarity between the predicted label distribution to the ground truth distribution. According to the definition of K-L divergences, the distance between two discrete probability distributions $P \in R^i, Q \in R^i$ is

$$
\begin{aligned}
D_{KL}(P\|Q) &= \sum_i P_i \log \frac{P_i}{Q_i} \\
&= \sum_i P_i \log(P_i) - P_i \log(Q_i).
\end{aligned}
\tag{1}
$$

In particular, given the training data with the Gaussian label distribution, after through the shared sub-network, an image m is mapped to a c-dimensional probability score $Q_m \in R^c$ ($Q_m^j = exp(f_m^j)/\sum_{k=1}^c exp(f_m^k)$), where f_m is the c-dimensional intermediate feature of the output of the shared sub-network for the image m and Q_m^j is the probability that image m is in age j. The loss for the image m is defined by

$$
minloss = \sum_c^{j=1} l_m^j \log(l_m^j) - l_i^j \log(Q_m^j) = \sum_c^{j=1} -l_m^j \log(Q_m^j).
\tag{2}
$$

We optimize the network parameters via back propagation. The gradient of the softmax function is

$$
\frac{\partial Q_m^j}{\partial f_m^j} = Q_m^j(1 - Q_m^j).
\tag{3}
$$

Here we provide the gradient of *loss* with respect to f_m^j:

$$
\begin{aligned}
\frac{\partial loss}{\partial f_m^j} &= \frac{\partial loss}{\partial Q_m^j} \cdot \frac{\partial Q_m^j}{\partial f_m^j} \\
&= -l_m^j \cdot \frac{1}{Q_m^j} \cdot Q_m^j(1 - Q_m^j) \\
&= Q_m^j - l_m^j.
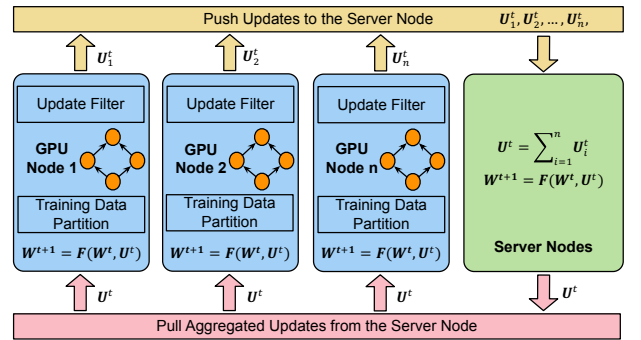\end{aligned}
\tag{4}
$$

Fig. 3. The system architecture of our proposed distributed model training system. Our system is designed based on the PF framework, which contains a set of GPU nodes and a server node. During the computation, each GPU node pushes generated updates U to the server node for aggregation, then pulls the aggregated updates to update the model parameters W. To reduce network overhead, we add a filter on each GPU node.

IV. SYSTEM OPTIMIZATION: COMMUNICATION EFFICIENT DISTRIBUTED MODEL TRAINING

In this section, we propose a method to reduce the communication overhead for the traditional PS-based distributed model training system, and implemente it into Caffe.

1) Distributed Model Training with the PS Framework: To handle large-scale DL applications, a set of distributed model training systems like TensorFlow have been proposed based on the PS framework to execute data-parallel ML algorithms. The PS framework contains a group of server nodes and a group of GPU worker nodes. The training model's parameters are globally shared and managed on the server nodes. Training dataset is partitioned and assigned to the GPU worker nodes. In this work, we only consider the case with multiple GPU worker nodes and one server node.

A data-parallel ML algorithm usually executes the following equation iteratively on the PS framework until some conver-
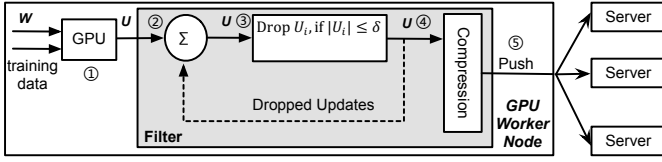
Fig. 4. Caffe-ASU allows GPU worker nodes to selectively drop updates with a given threshold during the push operations.

gence criteria are met:

$$\text{GPU Worker Nodes}: \quad U_i^t = \Delta(W^t, \mathcal{D}_i),$$
$$\text{Server Node}: \quad W^{t+1} = F(W^t, \sum_{i=1}^{n} U_i^t), \quad (5)$$

where i is the index of the i-th GPU worker node, W^t denotes the parameter vector at t-th iteration, U_i^t is the update vector computed by i-th GPU node at t-th iteration using the function $\Delta(\cdot)$ and input data set $\mathcal{D}_i$, and $F(\cdot)$ is the function used to update the model parameter vector using aggregated updates. During the training process, a GPU node continuously performs computation on W and outputs U, which is aggregated on the server node to update W. To exchange data between GPU nodes and the server node, the PS framework defines a push/pull communication model: GPU worker nodes push computed U to the *server* node, and pull latest W from it.

2) Caffe-ASU: We design a filter to allow each GPU node to selectively drop some entries of the update vector during the push operation to reduce network traffic and communication time. In the next push operation, dropped updates would be accumulated into the newly generated update vector. In this way, each GPU worker node would push aggregated sparse updates (ASU) to the server node, rather than push all updates in each iteration. We implement this file in Caffe-MPI, and name the system Caffe-ASU.

As shown in Fig. 4, before the push operations, Caffe-ASU selectively drops some updates as follows:

$$\begin{cases} U_{drop,i}^t = U_i^t, & \text{if } |U_i^t| \leq \delta \\ U_{rmn,i}^t = U_i^t, & \text{if } |U_i^t| > \delta \end{cases}, \quad (6)$$

where U_{rmn}^t denotes the updates that are pushed to the server node by i-th GPU node, $U_{drop,i}^t$ contains all dropped updates in this push operation, and δ is a predefined threshold. In the next communication operation, U_{drop}^t would be accumulated into the newly generated update vector as

$$U_i^{t+1} \leftarrow U_{drop,i}^t + U_i^{t+1}. \quad (7)$$

Since each GPU node just sends a partition of updates to the server node, Caffe-ASU would not update all parameters in each iteration. Therefore, during the pull operation, GPU worker nodes only pull updated parameters. Our experiments (see Section V) showed that this policy could reduce communication time by a factor of 30 without losing training progress or model quality.

V. EXPERIMENTS AND NUMERICAL RESULTS

In this section, we evaluate the efficiency and effectiveness of proposed intelligent demography system. The following describes the details of the experiments and results.

A. Age Model Performance Evaluation

1) Age Training Datasets: To make the age model more appropriate for the real world application, we set our experiments based on two latest in-the-wild aging datasets crawled from web, i.e. IMDB-WIKI and AgeDB dataset. The details are listed in Table I.

IMDB-WIKI dataset [23] is the largest publicly available dataset for age estimation of people in the wild containing 460,723 face images from 20,284 celebrities from IMDb and 62,328 from Wikipedia, thus 523,051 in total. According to the query list including the most popular 100,000 actors on the IMDb website, the profiles date of birth, name, gender and all images related to actors are crawled.

AgeDB dataset [24] is an in-the-wild dataset with large variations in pose, expression and illuminations. It contains 16,488 images of various famous people with accurate to the year, noise-free labels. Every image is annotated with respect to the identity, age and gender attribute. There exist a total of 568 distinct subjects. The average number of images per subject is 29. The minimum and maximum age are 1 and 101, respectively.

TABLE I
AN OVERVIEW OF THE USED DATASET FOR OFF-LINE AGE MODEL TRAINING

Dataset	# Images	# Subjects	# Year
IMDB-WIKI [23]	523,051	20,284	2015
AgeDB [24]	16,488	568	2017

2) Age Estimation Evaluation: To evaluate the performance of age estimation algorithm, we use the Mean Absolute Error (MAE) as the evaluation measures The MAE is calculated based on the average of the absolute errors between the estimated age and the ground truth (labeled age), which is represented as

$$MAE = \frac{1}{N} \sum_{n=1}^{N} \|l_n - y_n\|, \quad (8)$$

where l_n is the ground truth label of the nth image and y_n represents the estimated age based on the proposed framework. N is the total number of testing samples.

And in this work, we also consider another measurement specifically for the demography estimation. We evaluate the estimation accuracy of age group with different age gaps. The goal is to predict whether a person's age within some range instead of predicting the precise biological age. Because in demography analysis, the age structure of a population refers to the number of people in different age groups. Here we evaluate the performance of age group with different age ranges: 5 years, 10 years, 15 years and 20 years.

TABLE II
MAE COMPARISON ON AGEDB AND IMDB-WIKI DATASETS.

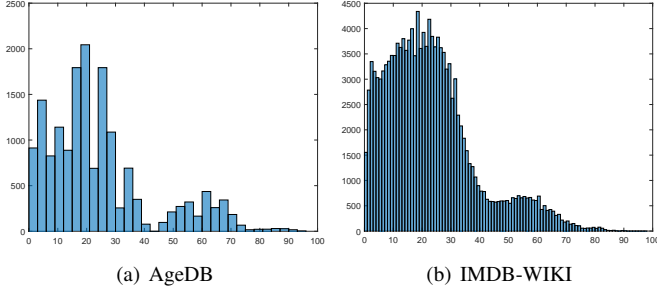| Dataset	Model	
	DEX [23]	Fine-tuned Model
IMDB-WIKI	38.4	**22.5**
AgeDB	28.47	**20.32**

(a) AgeDB

(b) IMDB-WIKI

Fig. 5. The MAE statistical results of age model testing on AgeDB and IMDB-WIKI dataset.

3) Age Estimation Performance: We first test the DEX model, which is pre-trained on the IMDB-WIKI dataset with softmax loss, on the whole AgeDB dataset. The MAE of estimation results is 28.47. Then we fine tune the whole network based on the AgeDB dataset by replacing the loss layer with K-L divergences distance and the MAE has dropped into 20.32. We also test the fine-tuned age estimation model on the IMDB-WIKI dataset and the MAE is reduced from 38.4 to 22.5. The detailed MAE comparison is listed in Tabel II. To be more clear illustration, we summarize the MAE distribution on the two aging datasets and show the result in Figure 5.

From this result we can see that the multi-age distribution label and the K-L distance loss can improve the wild age estimation significantly. Since the age estimation task for image in the wild is very challenging, We also list the age group accuracy in Table III.

TABLE III
THE AGE GROUP ESTIMATION OF DIFFERENT AGE RANGES.

Age Gap	Accuracy
5 years	23.6%
10 years	43.9%
15 years	62.3%
20 years	73.7%

B. Model Training Efficiency Evaluation

In this set of experiments, we measure the performance of Caffe-ASU. We use AgeDB as the training and test dataset, and run the experiments on 4 GPU virtual machines. In addition, we also implement Caffe-DSU based on [9], which uses a threshold to drop updates for pushing without aggregating dropped one. Caffe-RAW would push all updates and pull

(a) Percentage of Dropped Updates

(b) Test Loss

(c) Training Loss with Caffe-RAW

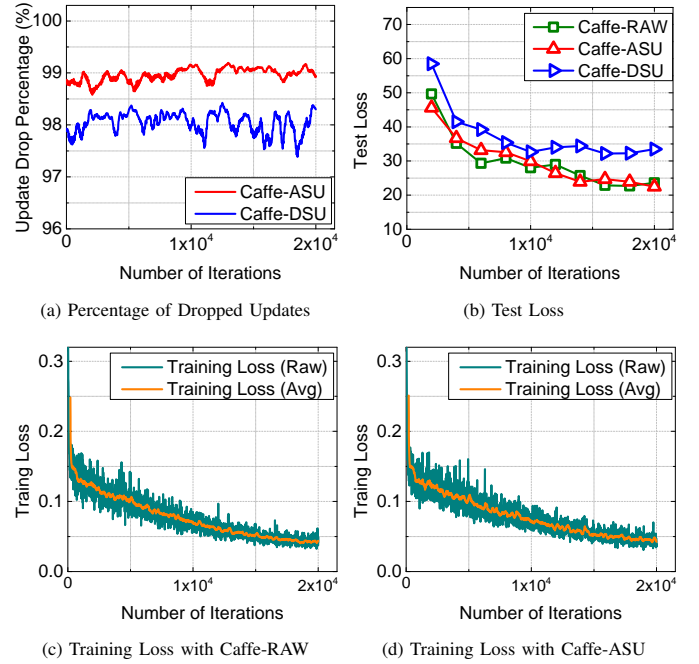(d) Training Loss with Caffe-ASU

Fig. 6. Performance of Caffe-ASU. (a) shows that percentage of dropped updates during the training progress. (b) shows the loss value on the test dataset. (c) (d) show the loss value on the training dataset using Caffe-RAW and Caffe-ASU, respectively.

all parameters at each iteration. In the experiments, we use 1×10^{-5} as the threshold, and users could select different values according to their particular applications.

1) Percentage of Dropped Updates: As shown in Fig. 6(a), when using 1×10^{-5} as the threshold, Caffe-ASU could drop about 98.8% of updates at each iteration. It means that Caffe-ASU pushes about 1.2% of updates to the server node, and pulls roughly 1.2% of updated parameters from the server node. In addition, our proposed deep learning model contains about 135 millions of parameters. When using Caffe-RAW, each GPU node needs to push about 540MB updates and pull 540MB parameters via network. As a comparison, Caffe-ASU allows each GPU node to push only 11MB and pull 15MB data on average in our testbed. We also note that Caffe-DSU could drop about 98% of updates at each iteration at the cost of losing model quality from Fig. 6(b).

2) Model Quality: Fig. 6(b) shows that Caffe-ASU would not reduce model quality, which is measured by the loss value on the testing dataset. Specifically, Caffe-ASU and Caffe-RAW could achieve the loss value of about 22.5 on the testing dataset. We can also find that Caffe-DSU could only achieve the loss value of about 32.2 on the testing dataset. It means that Caffe-ASU could drop about 98.8% of updates without losing model quality, which Caffe-DSU could not guarantee model quality if dropping about 98% of updates at each iteration. Thus, we could conclude that Caffe-ASU could achieve significant gains as compared to Caffe-DSU.

3) Training Progress: Fig. 6(b) shows that Caffe-ASU would not affect the training progress, which is measured by
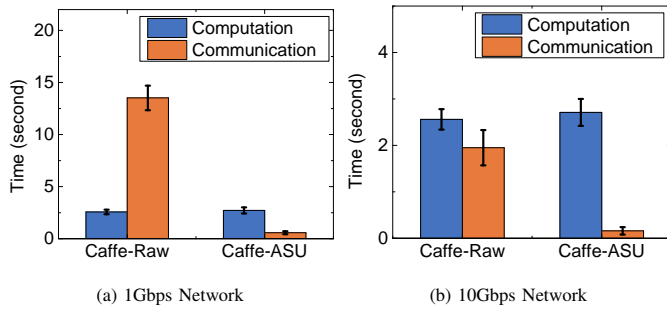
Fig. 7. Speedup ratio of Caffe-ASU using 1Gbps and 10Gbps network.

the loss value on the training dataset at different iterations. As we can see, at iteration 2×10^4, both Caffe-ASU and Caffe-RAW could achieve the loss value of about 0.038. It means that Caffe-ASU does not meed additional iterations to achieve the same loss value on the training dataset, compared with Caffe-RAW.

4) Speedup Ratio: We run the experiments using 1Gbps and 10Gbps network, and show the computation and communication time in Fig. 7. When using 1Gbps, Caffe-ASU could reduce the communication time by a factor of about 28 compared to Caffe-RAW, and speed up the training progress by a factor of about 4.9. When using 10Gbps, Caffe-ASU could reduce the communication time by a factor of about 16 compared to Caffe-RAW, and speed up the training progress by a factor of about 1.6.

VI. CONCLUSION

In this paper, we investigate the problem of speeding-up facial age estimation. We propose a high-efficient age estimation system with joint optimization of age estimation algorithm and deep learning system. The system is designed based on the three-tier fog computing architecture and provides the age group analysis directly from raw videos. Then we apply the system for intelligence demographics. Experimental results demonstrate the effectiveness and efficiency of our system. To our best knowledge, this is the first intelligent demographics system which implements the population investigation automatically via surveillance videos. In the future, we aim to further improve the performance of age estiamtion algorithm and apply the proposed system in the large-scale video surveillance of smart city project. The intelligent demographics system will improve the efficiency of smart cities and urban living.

VII. ACKNOWLEDGMENTS

REFERENCES

[1] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013, pp. 3476–3483.

[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 (CVPR), 2014 IEEE Conference on*, 2014, pp. 1701–1708.

[3] M. Yang, S. Zhu, F. Lv, and K. Yu, "Correspondence driven adaptation for human profile recognition," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 505–512.

[4] Z. Hu, Y. Wen, J. Wang, M. Wang, R. Hong, and S. Yan, "Facial age estimation with age difference," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3087–3097, 2017.

[5] Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua, "Ordinal regression with multiple output cnn for age estimation," in *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, 2016, pp. 4920–4928.

[6] S. Chen, C. Zhang, M. Dong, J. Le, and M. Rao, "Using ranking-cnn for age estimation," in *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, 2017, pp. 5183–5192.

[7] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*, 2014, pp. 675–678.

[8] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[9] M. Li, D. G. Andersen, A. J. Smola, and K. Yu, "Communication efficient distributed machine learning with the parameter server," in *Advances in Neural Information Processing Systems*, 2014, pp. 19–27.

[10] P. Sun, Y. Wen, T. N. Duong, and H. Xie, "Metaflow: a scalable metadata lookup service for distributed file systems in data centers," *IEEE Transactions on Big Data*, 2016.

[11] G. Gao, H. Hu, Y. Wen, and C. Westphal, "Resource provisioning and profit maximization for transcoding in clouds: a two-timescale approach," *IEEE Transactions on Multimedia*, vol. 19, no. 4, pp. 836–848, 2017.

[12] S. Yi, H. Li, and X. Wang, "Pedestrian behavior modeling from stationary crowds with applications to intelligent surveillance," *IEEE Transactions on Image Processing*, vol. 25, no. 9, pp. 4354–4368, 2016.

[13] C. W. Ling, A. Datta, and J. Xu, "A case for distributed multilevel storage infrastructure for visual surveillance in intelligent transportation networks," *IEEE Internet Computing*, 2017.

[14] A. Filonenko, K.-H. Jo *et al.*, "Unattended object identification for intelligent surveillance systems using sequence of dual background difference," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2247–2255, 2016.

[15] A. R. Alharbi and M. A. Thornton, "Demographic group prediction based on smart device user recognition gestures," in *Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on*, 2016, pp. 100–107.

[16] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition (CVPR), 2001 IEEE Conference on*, vol. 1, 2001, pp. I–I.

[17] M. Zaharia, T. Das, H. Li, T. Hunter, S. Shenker, and I. Stoica, "Discretized streams: Fault-tolerant streaming computation at scale," in *Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles*, 2013, pp. 423–438.

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2012, pp. 1097–1105.

[19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, 2015, pp. 1–9.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2016.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, 2016, pp. 770–778.

[22] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," *arXiv preprint arXiv:1608.06993*, 2016.

[23] R. Rothe, R. Timofte, and L. V. Gool, "Dex: Deep expectation of apparent age from a single image," in *IEEE International Conference on Computer Vision Workshops (ICCVW)*, December 2015.

[24] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou, "Agedb: the first manually collected, in-the-wild age database," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, 2017, pp. 1997–2005.