CURRICULUM AND ACTIVE SELF-PACED LEARNING WITH MINIMUM SPARSE RECONSTRUCTION FOR FACE RECOGNITION

A Thesis

by

Barış Büyüktaş

Submitted to the
Graduate School of Sciences and Engineering
In Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in the
Department of Electrical and Electronics Engineering

Özyeğin University
August 2020

CURRICULUM AND ACTIVE SELF-PACED LEARNING WITH MINIMUM SPARSE RECONSTRUCTION FOR FACE RECOGNITION

Approved by:

Prof. Tanju Erdem, Advisor
Department of Electrical and Electronics Engineering
Özyeğin University

Prof. Çiğdem Eroğlu Erdem, Co-Advisor
Department of Computer Engineering
Marmara University

Prof. Fatih Ugurdağ
Department of Electrical and Electronics Engineering
Özyeğin University

Asst. Prof. Furkan Kıraç
Department of Computer Science
Özyeğin University

Date Approved: 11 August 2020

Asst. Prof. Nihan Kahraman
Department of Electrical and Electronics Engineering
Yıldız Technical University

To my family and friends.

ABSTRACT

In this thesis, we present two different novel frameworks for face recognition. The first one is based on a novel curriculum learning (CL) algorithm for face recognition using convolutional neural networks. Curriculum learning is inspired by the fact that humans learn better, when the presented information is organized in a way that covers the easy concepts first, followed by more complex ones. It has been shown in the literature that CL is also beneficial for machine learning tasks by enabling convergence to a better local minimum. In the proposed CL algorithm for face recognition, we divide the training set of face images into subsets of increasing difficulty based on the head pose angle obtained from the absolute sum of yaw, pitch and roll angles. These subsets are introduced to the deep CNN in order of increasing difficulty. Experimental results on the large-scale CASIA-WebFace-Sub dataset show that the increase in face recognition accuracy is statistically significant when CL is used, as compared to organizing the training data in random batches.

The second framework can progressively train classifiers for face recognition when the available face images of subjects increase gradually. The framework starts with a small set of annotated images and includes new face images into the training set with minimum expert annotation effort using two different strategies: self-paced learning (SPL) and active learning with minimum sparse reconstruction (AL-MSR). SPL is used for automatic annotation of easy images, for which the classifiers are highly confident. AL-MSR is used to request the help of an expert for annotating difficult or low confidence images, which are further filtered for diversity using minimum sparse reconstruction. We combine self-paced and active learning with minimum sparse reconstruction in a single framework, namely ASPL-MSR. We improve the

recently proposed ASPL framework [1] by introducing MSR for eliminating "similar" images from the set selected by AL, which significantly reduces the required cost of expert annotation effort while providing a comparable recognition performance. The proposed framework is also robust against incorrectly annotated images in the training set. The proposed method (ASPL-MSR) is tested using two large-scale datasets (CASIA-WebFace and CACD) and promising results have been obtained. We can achieve the same face recognition accuracy by using less expert-annotated data as compared to the state-of-the-art.

ÖZETÇE

Bu tezde yüz tanımlama sistemleri için iki farklı yöntem sunulmaktadır. İlk yöntem, bilgilerin önce kolay kavramları, ardından daha karmaşık olanları kapsayacak bir müfredat dahilinde organize edildiğinde, insanların daha iyi öğrendikleri gerçeğinden esinlenmiştir. Literatürde müfredat öğrenmesinin (CL) makine öğrenmesi için daha iyi bir yerel minimum seviyeye yakınsama sağlayarak yararlı olduğu gösterilmiştir. Yüz tanıma için önerdiğimiz müfredat öğrenmesi algoritmasında, eğitim kümesinde yer alan resimlerdeki baş pozlarından elde edilen açıların mutlak değerlerinin toplamına göre alt kümeler elde edilerek, kolaydan zora açılar artacak şekilde sıralanmıştır. Bu alt kümeler, evrişimsel sinir ağının eğitimi sırasında artan zorluklarına göre sırayla verilmiştir. CASIA-WebFace-Sub veritabanındaki deneysel sonuçlar, müfredat öğrenmesi kullanıldığında doğruluğun, eğitim resimlerinin rastgele sıralanarak kullanılmasından daha başarılı sonuçlar verdiğini göstermiştir.

İkinci yöntem ise kişilere ait yüz resimlerinin zaman içinde arttığı durumlarda kademeli olarak yüz tanıma sisteminin eğitilmesine dayanır. İlk başta az sayıda etiketli eğitim verisi ile başlanır ve daha sonra gelen yeni veriler eğitim kümesine dahil edilirken etiketleme için en az insan gayreti kullanılması iki farklı yöntem ile sağlanır: kendi-hızında öğrenme (SPL) ve en az seyrek geri çatım ile aktif öğrenme (AL-MSR). SPL, sınıflandırıcının yüksek güvenirlikte olduğu kolay resimlerin otomatik olarak etiketlenmesine dayanır. AL-MSR ise düşük güvenilirlikte ya da zor olan ve birbirlerinden yeterince farklı olan resimlerin etiketlenmesi için uzman yardımına başvurur. SPL ve AL-MSR tek bir yöntem içinde birleştirilmiş ve ASPL-MSR olarak adlandırılmıştır. Daha önce önerilmiş olan ASPL yöntemi [1] en az seyrek geri çatım

(MSR) ile birbirinden farklı olan resimlerin seçilmesi için iyileştirilmiştir. Böylece uzman etiketlemesi için gereken çaba önemli ölçüde azaltılmış, öte yandan yüz tanıma performansı aynı kalmıştır. Önerilen yöntem, eğitim kümesindeki yanlış etiketlenmiş verilere karşı da dayanıklıdır. Önerilen ASPL-MSR yöntemi iki büyük veri kümesinde (CASIA-WebFace ve CACD) denenmiş ve olumlu sonuçlar elde edilmiş, literatüre göre daha az uzman etiketli veri ile aynı yüz tanıma başarımının elde edildiği gösterilmiştir.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor Prof. Çiğdem Eroğlu Erdem for providing guidance, patience, motivation and feedback throughout the project. She guided me so positively that increased my interest and enthusiasm for the topics I worked on.

I would like to thank my advisor Prof. Tanju Erdem for guiding my thesis with his valuable research ideas. His continuous support helped me writing of this thesis.

Besides my advisors, I would like to thank Prof. Fatih Uğurdağ, who has been a mentor to me since my undergraduate education. He has always trusted and supported me throughout my academic career.

I would like to thank the rest of my thesis committee; Asst. Prof. Furkan Kıraç and Asst. Prof. Nihan Kahraman for their challenging questions and insightful comments.

Very special thanks to my mother Mrs. Nuran Büyüktaş and my father Mr. Cengiz Büyüktaş, who has supported my decisions throughout my life. I also thank my aunt Asst. Prof. Derya Yılmaz Adalıoğlu for the true moral support whenever I needed.

This thesis was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under project 116E088. The Titan V used for this research was donated by the NVIDIA Corporation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Face recognition is an important problem, which has been attracting the attention of researchers for a long time [3, 4]. It has become prevalent in many application areas in the last decade, including security, health, law enforcement, entertainment, education and marketing [5]. Face recognition using facial images is a difficult computer vision and pattern recognition problem due to difficulties arising from varying head pose, illumination, age, facial expressions, facial hair and makeup [6].

During the past few years, great progress has been achieved for face recognition using deep convolutional neural networks (DCNN), which requires large annotated datasets for training containing a diverse set of images for each subject [7, 8, 9, 10, 11, 12, 13]. The diversity of face recognition datasets arise from variations in head pose, illumination, resolution, image quality and accessories [14, 15, 16, 17, 18, 19, 20]. Diversity of image datasets has been shown to increase the accuracy of DCNNs for classification tasks [21]. During the training phase, these images are fed to the network in batches, which are selected randomly from the training dataset and hence may contain both easy and more challenging samples of the same person. The CNNs learn the optimal parameters from these randomly formed batches of data.

In Chapter III, we propose a novel curriculum learning (CL) algorithm for face recognition. Curriculum learning was introduced to present the training data to deep neural networks in order of increasing difficulty, and was shown to improve generalization by enabling faster convergence to a better local minimum [22]. Curriculum learning was applied to several classification tasks [23, 24] and was recently applied successfully for facial expression recognition, where high intensity expressions were

considered as easy samples [25]. We investigate the idea that head pose can be used as a measure of difficulty, especially for large datasets collected in the wild with a variety of head poses. Hence, we present the training data to the CNN, in order of increasing difficulty using the head pose information obtained from pitch, yaw and roll angles. Experimental results on the large scale CASIA-WebFace dataset [14] indicate statistically significant improvements in face recognition accuracy using the proposed curriculum learning approach as compared to random ordering of training data. To the best of our knowledge, this is the first work, which applies curriculum learning to the task of face recognition.

Remarkable improvements have been achieved for face recognition in unconstrained environments using large scale datasets and deep convolutional networks [26, 27, 28]. The large scale datasets containing millions of annotated face images required to train DCNNs are very tedious, time consuming and costly to collect and annotate [20, 16, 8]. In some applications, the data may be available incrementally. That is, initially a small number of face images may be given for each person, and new images may become available incrementally in time. Therefore, it is very important to reduce the annotation effort for new incoming data and large datasets, while keeping the labelling errors at a minimum.

Another important problem with very large datasets is that they may include a considerable number of noisy labels. Many face images can easily be retrieved from the web with search engines using keywords. However, the keywords may not always reflect the true labels of the images. For instance, there are many misclassified face images in the CASIA-WebFace-Sub dataset [14, 1], an example of which is that two different identities playing in the same TV series have the same label. The cause of mislabeled images may also be human annotation error. These instances mislead the deep convolutional neural network and decrease the recognition accuracy. Therefore, it is important to detect and fix the incorrect labels.

Recently, an active self-paced learning (ASPL) framework has been proposed for progressive and cost-effective face identification [1]. This framework initially starts with a small set of annotated dataset and trains a preliminary classifier on this small set. Then, at the iteration learning stage, self paced learning (SPL) [29, 30, 31] and active learning (AL) [32, 33, 34] techniques are combined to incrementally expand the training data by selecting from incoming unlabeled data and then updating the classifiers using the expanded training dataset.

The idea of SPL is to automatically annotate the new unlabeled samples, which have the highest classification confidence using the current classifier. These high-confidence samples are "easy" samples for the classifier, and gradually more difficult samples are automatically labeled, as the classifier becomes more robust with increasing data. This can also be seen as a dynamic curriculum, which is similar to the learning process of humans. Humans start learning with easier concepts and then handle more difficult concepts gradually. The idea of active learning is on the other hand, to request expert (human) annotation for the most informative unlabeled images. The most informative samples are generally identified as the samples, which have the most uncertainty or low classification confidence using the current classifier. The low-confidence samples are in fact "difficult" ones for the current classifier. An illustration of high and low-confidence samples are given in Fig. 1, and the exact definitions will be given in the following sections. Although the AL and SPL approaches use conflicting criteria, they complement each other well and the annotation task is divided between the human and computer. The resulting ASPL system has been shown to achieve the same accuracy similar to using the whole annotated data, using only a small fraction annotated by hand [1].

In Chapter IV, we improve the ASPL framework by introducing a minimum sparse reconstruction (MSR) based image selection process to the AL step. The unlabeled images selected by the AL process (i.e. low-confident images) may be similar to

each other, which is not considered in ASPL [1]. Hence, instead of requesting expert annotation for all low-confident (uncertain) but possibly similar images, we select a diverse subset of representing them in the minimum sparse reconstruction sense. That is, data selected for human annotation should be the most uncertain and also sufficiently different from each other. In Fig. 1, an illustration of diverse and low-confidence samples is provided. For the subject on the left, the two images with sun glasses are both low-confidence samples, but they are similar to each other. Hence, it is not necessary to label both of them manually at first. Selecting one of them for human annotation should be sufficient for resolving the ambiguity about the identity. When the system is re-trained with newly added annotated samples and the classifiers become more mature, the other image with the sun glasses have the chance to be annotated automatically.

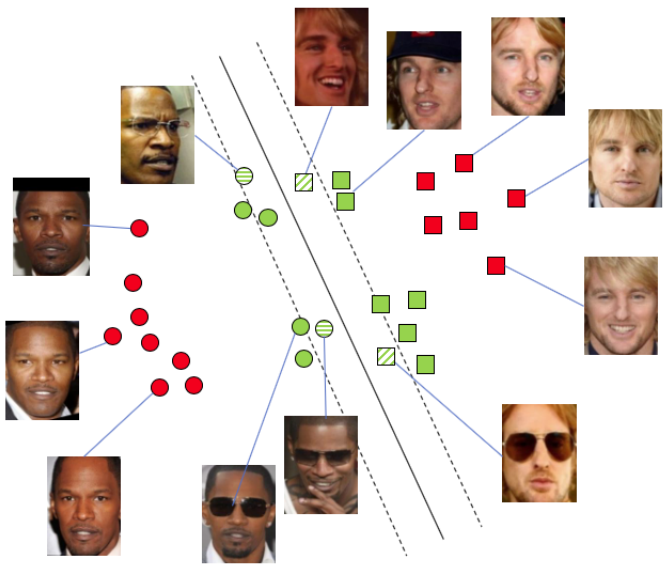

Figure 1: Illustration of high-confidence and low-confidence samples for two classes. Red dots represent high-confidence samples, which are located far from the decision boundary. Green dots represent low-confidence samples, which are located close to the decision boundary. The green dots with a stripe pattern are the diverse and low-confidence samples, which are determined by minimum sparse reconstruction.

This novel active learning with minimum sparse reconstruction (ASPL-MSR) approach further reduces the number of images to be annotated by the expert, hence decreasing the cost of the overall framework. We also observed that the automatic annotation can correct noisy labels. The experimental results on two large scale datasets (CASIA-WebFace-Sub [14, 1] and CACD [35, 1]) show that, ASPL-MSR can achieve the same (even slightly higher) accuracy using less number of expert-annotated images as compared to ASPL.

1.1 Contributions of the thesis

The two main contributions of the thesis can be summarized as follows:

- We propose a novel curriculum learning algorithm for face recognition. We show that head pose can be used as a measure of difficulty, especially for large datasets collected in the wild with a variety of head poses. The training data is presented to to the CNN in order of increasing difficulty, which indicate statistically significant improvements in face recognition accuracy as compared to random ordering of training data [36]. To the best of our knowledge, this is the first work, which applies curriculum learning to the task of face recognition.

- We improve the recently proposed active self-based learning framework [1] by introducing a minimum sparse reconstruction based diverse image selection process to the active learning step. This novel ASPL-MSR approach further significantly reduces the number of images to be annotated by the expert, hence decreasing the cost of the overall framework, while keeping the face recognition at the same level.

1.2 Organization of the thesis

The organization of the thesis is as follows. In the following chapter, we summarize the literature review of face recognition, curriculum learning, self-paced learning, active

learning and face recognition datasets. In Chapter III, we explain the motivation and methods of the curriculum learning for face recognition and show the results. In Chapter IV, we explain active learning, self-paced learning and minimum sparse reconstruction methods and present how these methods are combined to reduce the total annotated samples and show the experimental results with two different datasets. In Chapter V, we conclude the thesis and identify potential studies for future.

CHAPTER II

LITERATURE REVIEW

In this chapter, we provide a brief literature review about face recognition methods, curriculum learning, self-paced learning, active learning and face recognition datasets.

2.1 Face Recognition

Face recognition is a challenging computer vision and pattern recognition problem, which has been attracting the attention of researchers for a long time. Numerous methods have been proposed for face recognition, which can be broadly categorized as "shallow" or "deep" methods [8, 6]. Shallow methods use hand-crafted representations such as SIFT [37], SURF [38], and HOG [39]. Recently, great progress has been made as a result of the rapid developments in deep learning methods [40]. Utilization of deep learning methods accelerated after the work of Krizhevsky et al. [2], which demonstrated their effectiveness in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC)-2012 object recognition competition. Later on, deep CNNs were also used successfully for face recognition [7, 8].

In one of the first works, which apply deep CNNs for face recognition, Taigman et al. [7] used 4 million face images from a total of 4000 people to train a deep CNN. Features of the images were extracted using the deep CNN and then face pairs were compared using the Euclidean distance or chi-squared distance for face verification. In [9], triplet loss was used to maximize the distance between embeddings of a pair of images from different identities, while minimizing the distances between embeddings of images belonging to the same identity. Parkhi et al. [8] assembled a large scale face dataset (VGGFace) and investigated different CNN architectures showing comparable results with other state-of-the-art approaches using a single model. Wen et al. [41]

proposed a center loss rather than softmax loss to improve the discriminative power, and it achieved the state-of-the-art results on Labeled Faces in the Wild (LFW) and YouTube Faces (YTF) datasets. Ranjan et al. [10] added L_2-norm constraint to the softmax loss. Regularized loss increased the accuracy and achieved promising results on the IJB-A [42] dataset. Masi et al. [43] introduced pose-aware models, which uses pose-specific deep CNNs for training. Unlike the traditional face recognition methods, this approach utilized rendered half-profile and full-profile face images. Ge. et al. [44] proposed a two-stream convolutional neural network to identify low-resolution images with a low computational cost. One of the streams is a complex CNN, which is a highly-accurate model and the other is a simpler CNN for low-complexity. Zangeneh et al. [45] introduced an algorithm for the recognition of low resolution face images. In their approach, low and high-resolution face images were mapped into a common space. Deng et al. [12] enhanced the discriminative power of the face recognition model by using an Additive Angular Margin Loss. It outperformed the state-of-the-art face verification and face identification results on MegaFace Challenge1. Huang et al. [13] offered an approach for models, which are trained on imbalanced datasets. It combined the simple k-nearest cluster algorithm with Cluster-based Large Margin Local Embedding. A relatively small training dataset (CASIA-WebFace) was used for training and yet it reached the same accuracy as FaceNet [9].

Deep learning methods increase the face recognition performance significantly. However, larger variations in head pose reduce the performance [46]. In order to overcome this difficulty, larger face recognition datasets [14, 16, 17, 18, 19, 20] that consist of images with different yaw, pitch, roll angles emerged. In order to handle pose variatons, Ding et al. [47] proposed a trunk-branch ensemble CNN architecture, which combines features of the holistic face image and facial components by sharing some layers between CNNs. In another approach [48], 3D rendering was used to generate multiple synthetic face poses to be used for training separate CNN models

for three different head poses. Training separate DNNs for each head pose increased the complexity of the overall model.

In some cases, not all the face images are available at first. Images can join the training set after training begins. In such circumstances, dynamic model is required to incorporate new images into the model. Incremental face recognition proposed a solution to this problem. Incremental support vector machines (ISVM) [49] were used to update new samples with SVM classifier without re-training the whole dataset. Later, Ozawa et al. [50] proposed an approach that feature space was also learned in addition to classifier.

2.2 Curriculum Learning

The term 'curriculum' is defined as the educational process with regard to the educator's or school's instructional goals [51]. The educational process mostly starts with easier subjects. As the learning level increases, students can understand more complex subjects readily. This is due to the well-known fact that it is more intuitive for humans to start learning with easy samples at the beginning of learning. Elman [52] demonstrated this fact with children's language learning using training samples presented from easy to complex and in random order. Spitkovsky et al. [53] showed that the success of grammar induction is related to the length of the sentences. They proved that gradually increasing the complexity of the training set is beneficial to learning a grammar.

Curriculum learning has been used in machine learning applications for a while. Curriculum learning in machine learning mainly aims to improve the speed of convergence and achieve better generalization by increasing the complexity of the training data gradually [54]. Curriculum learning has been applied to various machine learning problems. Lapedriza et al. [55] ranked the training data from the most "valuable" to the least and showed that all the training samples are not equally important, hence,

performance of the model was improved by training a subset rather than the training whole dataset. Pentina et al. [23] used curriculum learning for object categorization. It was shown that learning multiple tasks in an order is more beneficial as compared to learning them jointly. Amramova [24] stated that ordering the training samples in ascending or descending order based on their difficulty could affect the convergence of a deep CNN.

Bengio et al. [56] used curriculum learning with recurrent neural networks to help the model explore further during training for the tasks of image captioning, constituency parsing and speech recognition. Guie et al. [57] demonstrated that curriculum learning is useful for facial expression recognition. They split their training data into subsets of increasing complexity levels based on the intensity of the facial expression. The training started from the easiest samples. As the network became more robust, complex samples were added to the training set without discarding easy samples.

2.3 Self-Paced Learning

It is well-known that humans learn better when easier concepts are introduced first, followed by more complex concepts. In education this is done by using a curriculum, which is designed by instructors. Inspired by this, Bengio et al. [54] proposed a curriculum learning method to show that it is also beneficial for machine learning tasks, which was later applied to several problems [23, 57, 36].

Later, curriculum learning inspired the emergence of self-paced learning. Kumar et al. [29] organized samples from easy to more complex to avoid bad local minima in the optimization process. However, the difficulty levels of the samples were not readily available at first. Hence, they proposed a self-paced learning algorithm to automatically select the easiest samples at each iteration. Lee et al. [30] introduced an approach for batch clustering, which focuses on easy samples first. The easiness

was determined based on the probability of the generic objectness and surrounding objects. Supancic et al. [31] used self-paced learning to select the easiest frames for object tracking. The model learned easiest frames before the more complex ones. When the model became more accurate, the images that used to be difficult became easier, hence, errors were corrected. Jiang et al. [58] proposed a method called self-paced learning with diversity, in which the model not only considers the easiness of samples but also tries to select diverse ones. In [59], Tang et al. used easy samples first to train a dictionary and added more complex samples iteratively for image classification. Hence, they obtained a better locally optimal dictionary. Jiang et al. [60] utilized self-paced learning for multimedia event search. They annotated samples automatically by re-ranking technique, and thus achieved state-of-the-art results on TRECVID. Zhao et al. [61] combined matrix factorization with SPL to find a better local minimum. Jiang et al. [62] stated that curriculum learning and self-paced learning have different learning schemes. Curriculum learning uses prior knowledge to specify the complexity of the samples. However, it neglects learner feedback. Therefore, they proposed an algorithm to link SPL and CL. Li et al. [63] improved the robustness of SPL by proposing a multi-objective self-paced learning method. This approach reduced the negative effects of bad initialization on the model.

The reason we use self-paced learning in our study is the same as the [1], which is to annotate unlabeled but easy samples automatically to reduce human effort.

2.4 Active Learning

Active learning tends to select the most informative samples and annotates them for faster convergence. It avoids to select redundant images, hence, the model achieves the same accuracy with less training data. Moreover, sometimes it is difficult or time-consuming to get the label of training samples for some supervised learning tasks, such as human activity recognition [64], speech recognition [65], visual tracking [66]. In

such cases, the importance of getting the high accurate network with less labeled samples increases.

There are several query strategies for specifying informative images. These are uncertainty sampling [67], query-by-committee [68], expected model change [69], expected error reduction [70], variance reduction [71] and density-weighted methods [71]. The most common one is uncertainty sampling. In that approach, labels of the least certain samples are asked to the expert. Initial classifiers are used to determine the informativeness of unlabeled samples. Different algorithms have been used to decide the uncertainty of the samples. In [32], posterior probability that closest to 0.5 was selected as most uncertain for binary classification. The purpose of the this work is reducing total amount of training data for text categorization. Scheffer et al. [33] used hidden markov models to learn classifiers, then difficult instances were selected using expectation-maximization algorithm. Tong et al. [34] proposed an algorithm that active learner queried the instances based on SVM decision boundary. Chang et al. [72] introduced active learning with SVM method, which achieves high accuracy and precision for image retrieval. Culotta et al. [73] proposed an algorithm for labelling instances that not only reduces the total number of annotated samples, but also takes into account the difficulty of labelling each sample. Kapoor et al. [74] used Gaussian Processes model to estimate uncertainty. Active learning was also used for object recognition in [75]. The most informative samples were determined based on the expected entropy of the unlabeled data. Joshi et al. [76, 77] used 2 most likely predictions to select uncertain images. These 2 approaches relied on the robustness of the network. Lin et al. [78] combined active learning with self-paced learning to increase the robustness of the classifiers. Our work was inspired by this work. In each iteration, high confidence images were added into training automatically, hence, classifier has become more reliable for active learning.

2.5 Face Recognition Datasets

The Labelled Faces in the Wild (LFW) dataset [79] was introduced in 2007, which contains 13,000 images from 5,749 subjects. It is the first dataset created for uncontrolled face recognition problems. Yi et al. [14] collected the face images in CASIA-WebFace dataset from Internet in a semi-automatical way. It contains 494,414 images from 10,575 people. VGGFace dataset [8] is a large scale public dataset, which has 2.6 million images from 2,622 people. MegaFace dataset [16] was introduced in 2016 to show the effects of gallery distractors on identification and verification. It has 4.7 million images ranged from 690,572 subjects. Another publicly available dataset named VGGFace2 was collected, which contains 3.31 million images with a large variation from 9131 identities [11]. In the thesis, we show the effectiveness of our methods using CACD and CASIA-WebFace-Sub datasets.

Table 1: Statistics for the commonly used face recognition datasets. The three entries represents the minimum-average-maximum images per subject.

Datasets	Number of subjects	Number of images	Number of images per subject	Year
LFW [79]	5,749	13,233	1-2.3-530	2007
CASIA-WebFace [14]	10,575	494,414	2-46.8-804	2014
CACD [35]	2,000	163,446	22-81.72-139	2014
Google [9]	>10M	>500M	50	2015
VGGFace [8]	2,622	2.6M	1,000	2015
MegaFace [16]	690,572	4.7M	3-7-2469	2016
IJB-C [19]	3,531	31,334 images, 11,779 videos	36.3	2018
VGGFace2 [20]	9,131	3.31M	80-362.6-843	2018

Cross-Age Celebrity (CACD) dataset is a publicly available large-scale dataset, which contains more than 160,000 images from 2000 celebrities ranging in age from 16 to 32. Chen et al. [35] manually annotated subset of 200 celebrities. To increase the number of annotated samples and to compare our results with [1], we used the augmented version of the subset, which was generated by Lin et al. [1]. They manually annotated 300 celebrities, hence, subset contains 56,105 images from 500 celebrities.

These images are RGB with a size of 150×200 pixels.

Although the number of images per person ranged from 3 to 804 in CASIA-WebFace database, in order to reduce the imbalance in this distribution, individuals with fewer than 100 images were not used in the experiments as in [1]. Therefore, the CASIA-WebFace-Sub dataset used in the experiments contained 181,279 images from 923 individuals. These images are grayscale with a size of 100 × 100 pixels.

CHAPTER III

CURRICULUM LEARNING FOR FACE RECOGNITION

In this chapter, we present a curriculum learning algorithm for face recognition using CNNs. Training set of face images are divided into subsets of increasing difficulty based on the absolute sum of yaw, pitch and roll angles. These subsets are introduced to the deep CNN from easy to difficult.

Below, we first describe the head pose estimation method and then give the details of the proposed curriculum learning algorithm for face recognition, which is followed by experimental results.

3.1 Determining Training Data Difficulty Based on Head Pose

The effect of image quality on face recognition performance has been analyzed by Dutta et al. in [80]. Differences in area under the ROC curve were calculated for each image quality parameter, which were based on the head pose, illumination, noise, motion blur, and resolution. According to the experimental results, head pose was found to be the most important factor. Therefore, we decided to specify the complexity of images in our training set using the head pose angles. Since it is intuitive that upright frontal faces are the easiest to recognize, we used the sum of absolute values of yaw, pitch and roll angles of the head pose to rank the difficulty level of the face images in the training dataset.

There are a number of head pose estimation approaches in the literature [81, 82, 83, 84, 85]. We used OpenFace 2.0, which has been shown to perform well under difficult conditions [86].

CASIA-WebFace-Sub dataset contains face images taken with a variety of head

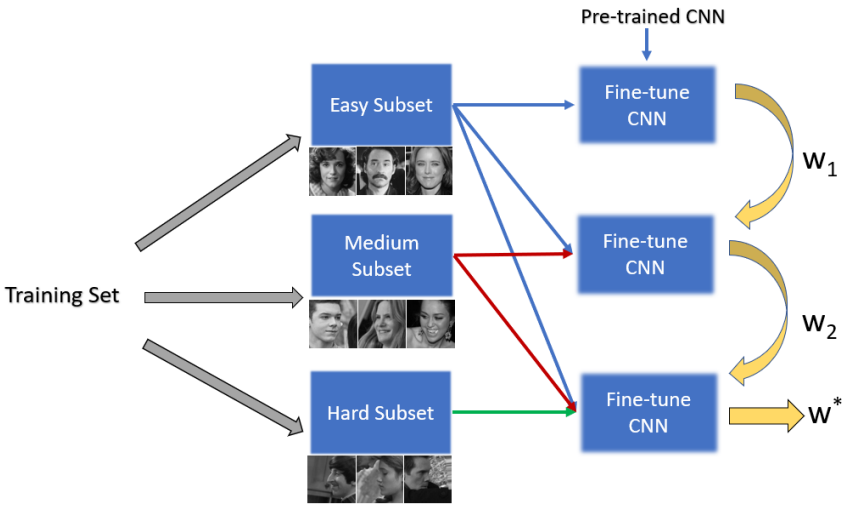

Figure 2: The framework of the proposed curriculum learning approach for face recognition. The training set is ranked into subsets using difficulty levels based on head pose. First, the easiest subset is used to fine-tune a pre-trained CNN. Then, more difficult subsets are included to the training set for further fine-tuning the CNN.

poses, several examples of which are shown in Fig.3 along with the estimated head pose angles and difficulty levels. The histograms of the absolute values of yaw, pitch and roll angles of the head pose are given in Fig.4, together with the histogram of their absolute sum. A yaw angle of 90°means, the face is seen from a profile view. The range of values for pitch and roll angles are less than 90°. As can be seen from the histogram of sum of absolute values of angles, most face images have a total angle below 50°, and less than 10% of the images have a total angle above 100°. We used the total angle value to rank the face images into subsets of increasing difficulty levels. Since the distribution of pitch and roll angles have a narrower range, yaw angle is the most effective angle for determining the total head pose angle.

3.2 Curriculum Learning for Face Recognition

During the training phase of a deep neural network, the weights of the model are learned using iterative numerical methods, since the objective function to be minimized is highly non-convex. Therefore, in curriculum learning, it is important to

Yaw	8.37°	14.4°	35.10°	49.14°	65.43°	68.22°
Pitch	0.63°	6.12°	23.58°	11.70°	9.45°	19.90°
Roll	1.62°	1.26°	0.81°	0.63°	11.70°	13.50°
Sum	10.62°	21.78°	59.49°	61.47°	86.58°	101.62°
Difficulty	easy	easy	medium	medium	hard	hard

Figure 3: Absolute values of estimated yaw, pitch, and roll angles of sample images from the CASIA-WebFace-Sub dataset. Difficulty level is determined by the sum of all absolute angles.

present the network with easy instances first, so that the network is guided towards a better local minimum. A recent theoretical analysis of curriculum learning shows that CL effectively modifies the optimization landscape [87].

We propose a novel curriculum learning algorithm for face recognition, which uses the intuitive difficulty in face datasets, the head pose. We propose to organize the training dataset into easy, medium and hard subsets using the total head pose angle. Existence of images with larger variations in head pose in the training set increases face recognition accuracy [88], but these difficult images should be introduced to the network after the easier ones.

We begin with the pre-trained AlexNet CNN model, which was trained on the ImageNet dataset for object recognition [2]. Then, we fine-tune the several final layers of the CNN using the easy subset (see Fig.1). After convergence, the medium subset is augmented to the easy subset and the fine-tuning process continues with the augmented training set until convergence. Finally, we augment the dataset with the hard subset and repeat. We summarize the followed steps in Algorithm 1.

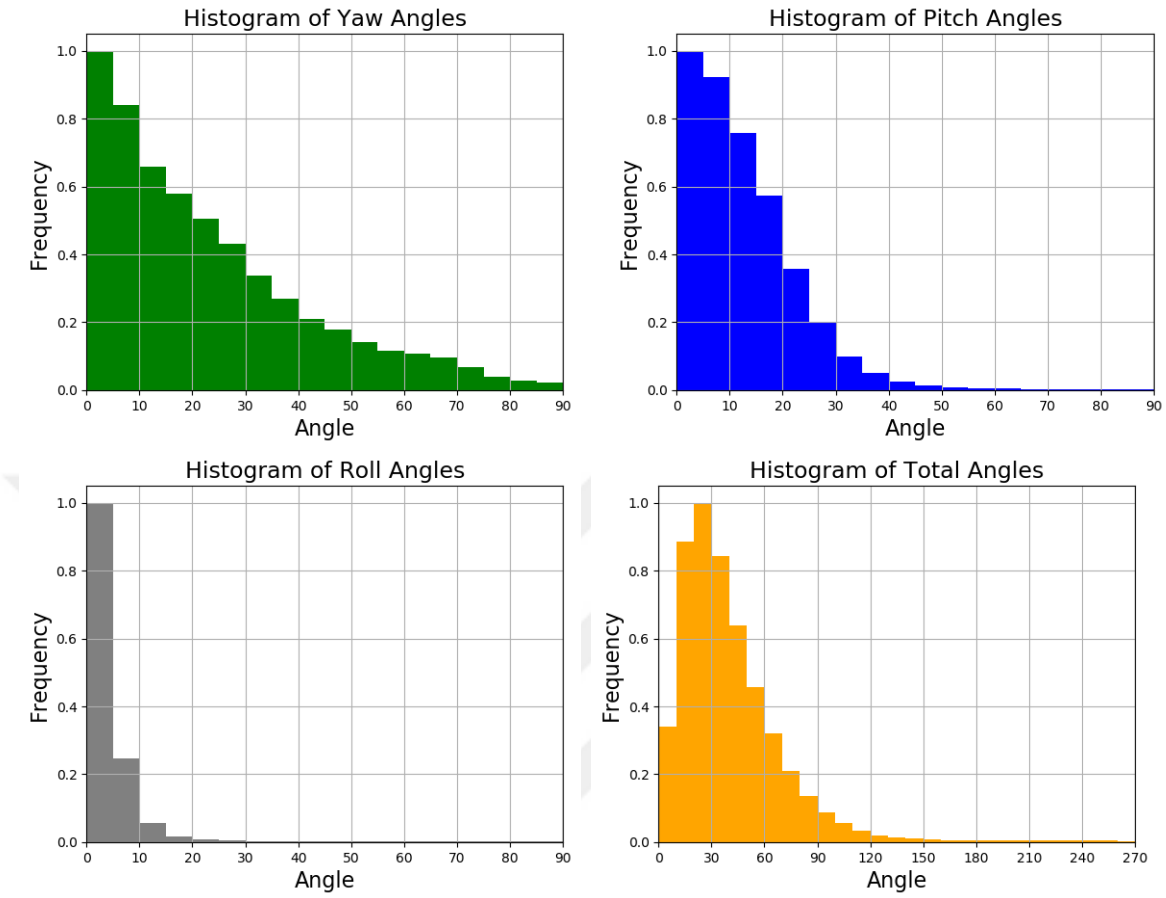

Figure 4: Histograms of absolute values of estimated yaw, pitch, roll and total angle for the images in CASIA-WebFace-Sub dataset.

3.3 AlexNet Architecture

Krizhevsky et al. [2] proposed a CNN model called AlexNet and won the visual object recognition challenge (ILSVRC) in 2012. There are five convolutional layers and 3 fully connected layers in AlexNet architecture. Rectified Linear Unit (ReLU) is used as an activation function (f(x) =max(0,x)). This function turns negative input values to '0'. The architecture of AlexNet can be seen in Fig. 5. In the thesis, we use the parameters of the pre-trained AlexNet in the first layers. AlexNet has sixty one million parameters. The vast majority of the parameters are held in last 3 fully connected layers. In [2], layers are divided into two; therefore, training could

Algorithm 1: Curriculum learning for face recognition

Inputs: Pre-ordered input dataset ranked into n subsets of increasing difficulty $X = \{X^i\}_{i=1}^{n}$, which is determined with the curriculum.
Pre-trained CNN.
Output: Optimal model parameters W*
$X^{train} = \emptyset$
for $i = 1 : n$ **do**
  $X^{train} = X^{train} \cup X^i$
  while *not converged* **do**
    fine-tune (W, X^{train})
  end
  choose optimal W*
  decrease learning rate
end

be completed with two GPUs. The input images with a size of 227x227x3 are passed through the first convolutional layers, which includes 96 filters with a size of 11×11 and a stride of 4. The second convolutional layer has 256 filters with a size of 5×5 and a stride of 1. The next 3 convolutional layers use the same filters with a size of 3×3 and a stride of one. The only difference is the total number of filters used. The third and fourth convolutional layers have 384 filters and the fifth convolutional layer has 256 filters. The first and the second fully connected layers have 4096 neurons and the last fully connected layer has 1000 neurons, which is the total number of classes on the ImageNet dataset. The main purpose of using this model in the thesis is that it has a relatively smaller architecture compared to other CNN models. In this way, we have saved time spent on training. It is possible to obtain more accurate network using more complex CNN models. However, our main goal is to reduce the total annotated samples instead of obtaining a high accurate model. In this chapter, we gave the results of fine-tuning the last, last two and last three fully connected layers when the rest of the parameters are frozen, which means that backpropagation does not reach frozen layers.

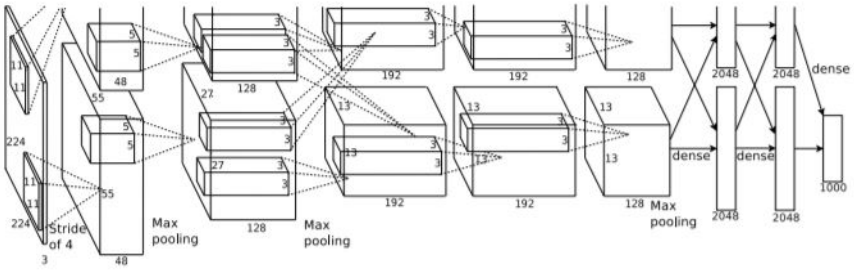

Figure 5: An illustration of the architecture of AlexNet [2].

3.4 Experimental Results

In this section, we first give the details of the experimental setup including the five datasets used in the experiments. Then, we compare the face recognition accuracies for three different CNN fine-tuning strategies using random ordering, curriculum-based ordering and inverse-curriculum based ordering of training data.

3.4.1 Experimental Setup

We used five-fold cross validation during the experiments. In each fold, 80% of the images (approx. 145.055 images) in CASIA-WebFace-Sub dataset were used for training and 20% of them were used for testing (approx. 36.255 images). Then, we estimated the head pose of all images in the training dataset and formed subsets of increasing difficulty. Head pose of 6464 out of 145.055 images could not be estimated with OpenFace 2.0, since they were blurry, noisy or non-frontal. Hence, these images are directly placed into the most difficult subset. Table 1 shows the total absolute head pose angles corresponding to the complexity (difficulty) ranking in the ordered training dataset.

We formed five different training image sets, from the whole training set of 145.055 images as shown in Table 2, which have increasing number of total samples selected from the whole training dataset. Image sets 1-3 consist of 30.000, 60.000, and 90.000 images, respectively. The goal of constructing image sets 1-3 is to study how the recognition accuracies are affected as the training dataset includes distinct difficulty

Table 2: Rank of the ordered images in the training set from easy to difficult and the corresponding total head pose angles.

Complexity Order	Angle
0	0.18°
10.000	10.53°
20.000	15.21°
30.000	19.08°
75.000	36.72°
85.000	41.76°
95.000	47.61°
105.000	54.81°
115.000	64.44°
125.000	80.28°
135.000	197.10°

levels between images in the easy, medium and difficult (hard) subsets.

- **Image set 1**: Consists of image subsets having a rank between 0-10.000 (very easy), 85.000-95.000 (medium), and 135.000-145.000 (very hard) in the ordered list of all images CASIA-WebFace-Sub dataset.

- **Image set 2**: Consists of subsets having a rank between 0-20.000 (easy), 80.000-100.000 (medium), and 125.000-145.000 (hard) in the ordered list of all images CASIA-WebFace-Sub dataset.

- **Image set 3**: Consists of subsets having a rank between 0-30.000 (easy), 75.000-105.000 (medium), and 115.000-145.000 (hard) of all images CASIA-WebFace-Sub dataset.

- **Image set 4**: Consists of all the training images in the CASIA-WebFace-Sub dat set. There are three subsets of increasing difficulty as follows. Images with a total absolute head pose angle below 35° were grouped as easy images, whereas images with angles between 35°- 65° and higher than 65° were grouped as medium difficulty and hard images, respectively.

- **Image set 5**: Consists of the same total number of images as image set 4, but it has been split into 4 subsets of increasing difficulty to provide a smoother transition from easy to difficult images. It is separated into subsets using total head pose angles as follows: 0° - 30° (Easy), 30° - 60° (Medium 1), 60° - 70° (Medium 2) and above 70° (Hard).

Table 3: Image sets obtained using the CASIA-WebFace-Sub dataset, and the number of images in each category.

	Total	**Easy**	**Medium**	**Hard**
Image Set 1	30.000	10.000	10.000	10.000
Image Set 2	60.000	20.000	20.000	20.000
Image Set 3	90.000	30.000	30.000	30.000
Image Set 4	145.055	71.150	44.355	29.550

	Total	**Easy**	**Medium 1**	**Medium 2**	**Hard**
Image Set 5	145.055	59.165	31.467	28.643	25.780

When we use curriculum learning, training starts with the easiest subset, and the next subset is augmented to the previous one after convergence. After each data augmentation step, the network is trained with a lower learning rate. Early stopping is used, when the accuracy of the network does not increase after 30.000 iterations with a test interval of 10.000 iterations. CNN is updated using the stochastic gradient descent with momentum 0.9, learning rate 0.001, and weight decay 0.0001.

3.4.2 Face Recognition Results

We first compared the performances of two AlexNet models without using curriculum learning: i) trained from scratch and ii) after fine-tuning the last two fully connected layers of the pre-trained AlexNet model using all 145.055 images in the training dataset. The accuracy after training from scratch was 77.28%, and the accuracy after

fine-tuning the last 2 fully connected layers was 80.93%. Hence, we observed that fine-tuning increased the accuracy by 3.65%.

We conducted experiments on the five training sets described above using three different approaches: with curriculum learning, without CL (i.e.random ordering) and with inverse CL (from hard to easy). We also investigated the effect of fine-tuning the last, last two and last three fully connected layers, which are given in Table 4, Table 5, and Table 6, respectively.

Table 4: Accuracies without CL (w/o CL), with CL (w/ CL), and with inverse CL learning, when fine-tuning the last fully connected layer.

Methods / Image sets	w/o CL	w/ CL	Increase	Inverse CL
Image set 1	0.4892	0.5232	3.40%	0.3631
Image set 2	0.6359	0.6894	5.35%	0.5838
Image set 3	0.7030	0.7482	4.52%	0.6743
Image set 4	0.7981	0.8106	1.25%	0.7513
Image set 5	0.7981	0.8135	1.54%	0.7682

Table 5: Accuracies without CL (w/o CL), with CL (w/ CL), and with inverse CL learning, when fine-tuning the last two fully connected layers.

Methods / Image sets	w/o CL	w/ CL	Increase	Inverse CL
Image set 1	0.5093	0.5396	3.03%	0.3876
Image set 2	0.6493	0.6971	4.78%	0.5942
Image set 3	0.7054	0.7490	4.36%	0.6727
Image set 4	0.8093	0.8223	1.30%	0.7613
Image set 5	0.8093	**0.8240**	1.47%	0.7787

Table 6: Accuracies without CL (w/o CL), with CL (w/ CL), and with inverse CL learning, when fine-tuning the last three fully connected layers.

Methods / Image sets	w/o CL	w/ CL	Increase	Inverse CL
Image set 1	0.4826	0.5238	4.12%	0.3894
Image set 2	0.6514	0.6963	5.35%	0.6013
Image set 3	0.7049	0.7488	4.49%	0.6784
Image set 4	0.8030	0.8160	1.30%	0.7523
Image set 5	0.8030	0.8181	1.51%	0.7785

According to the results given in Table 3, Table 4, and Table 5, the proposed CL algorithm consistently gives higher results as compared to random ordering. The highest accuracies are achieved when the last two fully connected layers are fine-tuned as can be seen in Table 4. Total accuracy increase on the five image sets are 3.03%, 4.78%, 4.36%, 1.30%, and 1.47%, respectively. The highest accuracy achieved is 0.8240 with CL on image set 5, which contains image subsets of 4 difficulty levels.

The inverse CL results are worse than random ordering of training data, which further justifies that ordering the training data from easy to difficult is indeed helpful for convergence to a better local minimum. Results on image set 5 show that using four subsets of difficulty is slightly better than using three subsets. First 3 image sets include the easiest and the most difficult samples. Therefore, the increase in accuracies when CL is used is higher for these datasets as compared to image sets 4 and 5.

We also used McNemar's Test [89] to investigate whether the increase in the accuracies are statistically significant. We used the number of misclassified examples with CL and random ordering using image set 5. Chi-square statistic with one degree of freedom was used. We obtained a score of 95.70, which indicates statistical

significance since it is above 3.84.

We also observed that the increase in accuracies with CL are due to the correction of the labels of images with higher difficulty levels. In Fig.6, histograms of the total head pose angles of all the test images and images whose labels have been corrected with curriculum learning are given. The average total head pose angle of all test images is 48°and the average total head pose angle of corrected images with CL is 58°. Therefore, we can conclude that using curriculum learning enables the CNN to recognize more difficult images better.

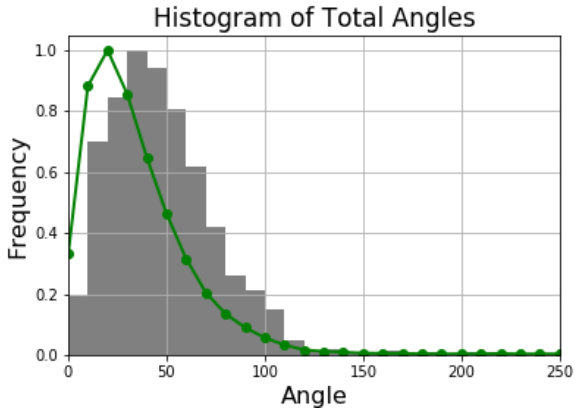

Figure 6: Histogram of total head pose angles for all test images (green curve) and images whose labels have been corrected with curriculum learning (gray bars).

In order to compare our results with the state-of-the-art, We used the same training parameters as [1] on the CASIA-WebFace-Sub dataset and followed a similar experimental setup. However, we don't restrict the total number of iterations during CNN fine-tuning. Instead, we waited until the accuracy doesn't change for 30.000 iterations. In [1], using all of the 145.055 training images the accuracy was approximately 0.77. Our curriculum learning based face recognition method gives an accuracy of 0.8240, which is significantly higher. Summary of our experimental results are given in Table 7.

Experiments were run on a workstation with an i7 3.4 GHz CPU, NVIDIA Titan V and NVIDIA GTX 1080 Ti GPUs.

Table 7: Summary of the face recognition results and comparison with the state-of-the-art.

Our Work (one layer)	0.8135
Our Work (two layers)	0.8240
Our Work (three layers)	0.8181
Lin [1]	≈ 0.77

3.5 Summary

In this chapter, we presented a novel curriculum learning algorithm for face recognition. In curriculum learning, the trained data is ranked into subsets of increasing difficulty, which was achieved using the total head pose angle. Experimental results on the large-scale CASIA-WebFace-Sub dataset shows that the recognition accuracy increases in a statistically significant way when the proposed CL algorithm is used as compared to random ordering of data when training a CNN. The increased recognition accuracy is especially evident in difficult images.

CHAPTER IV

ACTIVE SELF-PACED LEARNING (ASPL) WITH MINIMUM SPARSE RECONSTRUCTION (MSR) FOR FACE RECOGNITION

In this chapter, we present a framework that can progressively train classifiers for face recognition as the subjects' available face images gradually increase. Initially, we assume that a small set of images are annotated. SPL and active learning with minimum sparse reconstruction (AL-MSR) strategies are used to label new face images. We improve the ASPL framework [1] by introducing a MSR based diverse image selection process to the AL step.

Below, we first give an overview of the framework, and then give some background information about the formulation of SPL and ASPL. Then, the details of the proposed ASPL-MSR method is given, followed by experimental results.

4.1 Overview of the Framework

In this section, we give an overview of the proposed incremental face recognition framework, which is illustrated in Fig. 7. The overall framework consists of four main steps: feature extraction, classification, annotation of unlabeled data, updating the classifier and the feature extractor. Each step is described briefly below.

- *Feature Extraction*: We used learning-based features of images extracted using a deep convolutional neural network, since they provide features which have been shown to have better discrimination properties for face recognition [90]. Similar to ASPL [1], we used the pre-trained AlexNet model [2]. AlexNet was initially fine-tuned using 10% of the face dataset, which was randomly selected

for each person and manually annotated. Twenty percent of the rest of the data is reserved for testing and the other 80% is considered to be the unlabeled dataset.

- *Classification*: We used one-vs-rest linear SVM for classification. Initially, a classifier is trained for each person with the features extracted using 10% of data. These classifiers are later updated as the training data is augmented with incoming data.

- *Annotation of unlabeled data*: The current feature extractor and classifiers are used to classify the unlabeled images. Based on the confidence-level of each sample, either SPL or AL-MSR is used. If an image has been classified with high-confidence, it is **automatically annotated** using SPL and added to the labeled dataset. On the other hand, if an image has been classified with low-confidence, it is passed through a similarity detection step using MSR and then sent to the human expert for **manual annotation**.

  The confidence level of each image is determined by the distance of its feature vector to the decision boundaries of one-vs-rest SVMs. If a feature vector is further from the decision boundary, its confidence level increases, which will be described in more detail in the next section. For example, in Fig. 7, the feature vectors shown with red dots illustrate high-confidence features (images) since they are far from the linear decision boundary and the green dots illustrate low-confidence features (images) since they are closer to the linear decision boundary. Note that some green points are close to each other, indicating that they carry similar information. Hence, we can further subsample them to generate a representative dictionary and manually annotate only the samples in the dictionary.

- *Updating the classifier and feature extractor*: The classifiers are updated after

several runs of unlabeled data annotation step with the augmented training dataset. After several steps of classifier update, the DCNN used for feature extraction is also fine-tuned using the softmax loss.

The above process continues until all unlabeled images have been annotated.

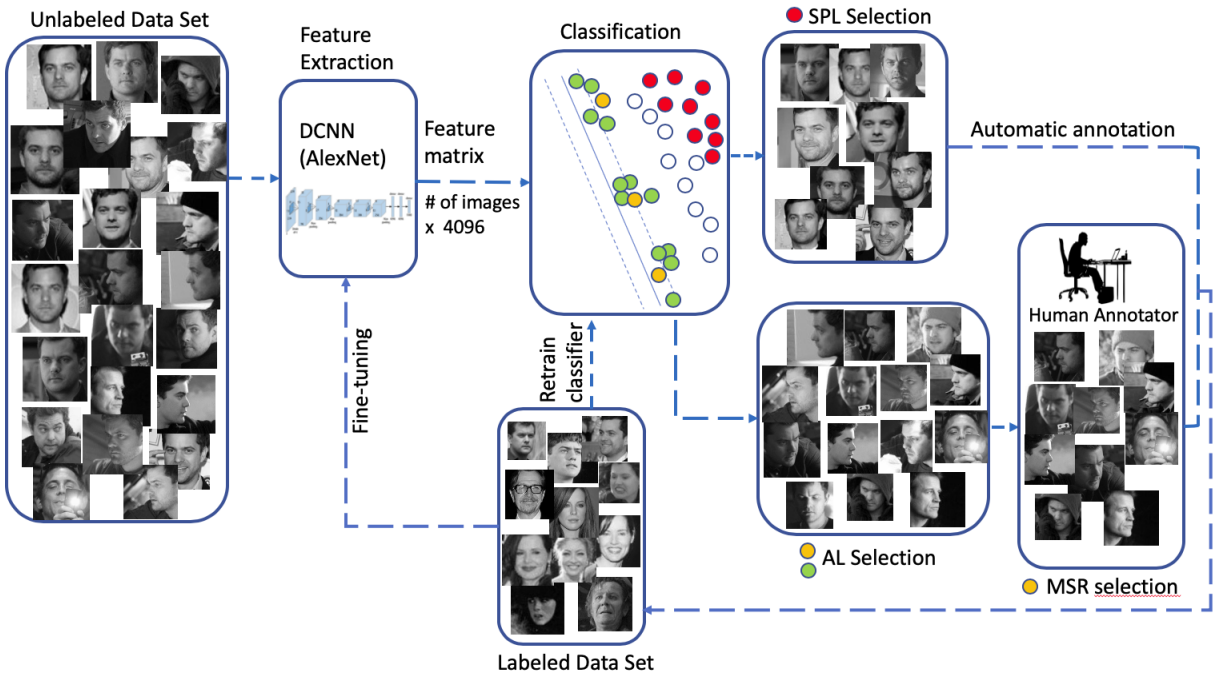

Figure 7: Illustration of the proposed ASPL-MSR method. The main steps are: feature extraction, classification, annotation of unlabeled data (using SPL and AL-MSR, updating the classifier and feature extractor.

4.2 Method

In this section, we give the details of the proposed active self-paced learning with minimum sparse reconstruction method for face recognition. We first explain how the problem is formulated and then explain each component of the algorithm, including automatic annotation of high-confidence samples using self-paced learning, expert annotation after active learning and minimum sparse reconstruction based uncertainty sampling.

4.2.1 Background

Before introducing our contribution, we would like to briefly review the formulation of self-paced learning, self-paced curriculum learning and active self-paced learning.

4.2.1.1 Self-paced Learning

In self-paced learning, the curriculum is dynamically generated by the learner, in contrast to using a static curriculum and has been formulated as an optimization problem by embedding the curriculum design as a regularization term into the learning objective function as follows [29, 62]. That is, easy samples are considered to be the ones, which can be classified with high-confidence.

Let us denote the training dataset as $D = \{(\mathbf{x}_i, y_i)\}$, $(i = 1, \ldots, n)$, where x_i is the d-dimensional feature representation of the i^{th} sample, and y_i is the corresponding ground truth label. Let $L(y_i, h(\mathbf{x}_i, \mathbf{w}))$ denotes the cost between the true label y_i and the estimated label $h(\mathbf{x}_i, \mathbf{w})$, and $\mathbf{w}$ denotes the model parameter of the learner with the decision function h. In SPL, the goal is to jointly learn the model parameter $\mathbf{w}$, and the latent binary variables v_i, which indicate whether the i^{th} sample is easy or not. The function to be minimized is:

$$\min_{\mathbf{w},\mathbf{v}} \mathbb{E}(\mathbf{w}, \mathbf{v}; \lambda) = \sum_{i=1}^{n} v_i L(y_i, h(\mathbf{x}_i, \mathbf{w})) - \lambda \sum_{i=1}^{n} v_i, \tag{1}$$

where $\mathbf{v} = [v_1, \ldots, v_n]^T$, that is $\mathbf{v} \in [0, 1]^n$. λ is a parameter for controlling the learning pace. The second term in (1) is a negative l_1-norm regularizer as $-\|\mathbf{v}\|_1 = -\lambda \sum_{i=1}^{n} v_i$ since $v_i > 0$. If λ is small, easy samples are the ones which give a small value of $L(.)$. Note that, samples are tied together in function $\mathbb{E}$ though the parameter $\mathbf{w}$, hence no single sample is easy, but a set of samples is considered to be easy if a parameter $\mathbf{w}$ can be learned such that the corresponding losses $L(.)$ are small. The problem in (1) can also be relaxed so that v_i is allowed to take any value in $[0, 1]$.

If $L(.)$ is convex in $\mathbf{w}$, (1) becomes a biconvex optimization problem with two sets

of variables $\mathbf{w}$ and $\mathbf{v}$, so that when one set is fixed, the optimal value of the other set can be solved by using convex optimization. In particular alternative convex search (ACS) [91] can be used to solve (1). When $\mathbf{v}$ is fixed, the optimal parameter $\mathbf{w}^*$ can be solved, and when $\mathbf{w}$ is fixed, the global optimum $\mathbf{v}^* = [v_1^*, \ldots, v_n^*]^T$ can be calculated by [29]:

$$v_i = \begin{cases} 1, & L(y_i, h(\mathbf{x}_i, \mathbf{w})) < \lambda \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

The above alternative search strategy can be interpreted as follows. When updating $\mathbf{v}$ with fixed $\mathbf{w}$, samples with loss values less than a threshold λ are considered as easy (or high-confidence) and will be added to the training dataset. When updating $\mathbf{w}$ with $\mathbf{v}$ fixed, the classifier will be trained on the training dataset augmented with the newly selected easily samples. The parameter λ can also be considered to the "age" of the model. If λ is small (model is young) only samples with small losses will be included to the training set, which are easier to learn. As the model matures, λ may increase to include samples with larger losses.

4.2.1.2 Self-paced Curriculum Learning

In [62], both ideas of static and dynamic curriculum have been combined and named as self-paced curriculum learning. This gives a "instructor-student collaborative" strategy, using the prior knowledge of the instructor as the curriculum and lets the student adjust to the curriculum dynamically. The function to be optimized is

$$\min_{\mathbf{w},\mathbf{v}} \mathbb{E}(\mathbf{w}, \mathbf{v}; \lambda, \Psi) = \sum_{i=1}^{n} v_i L(y_i, h(\mathbf{x}_i, \mathbf{w})) - f(\mathbf{v}; \lambda), \tag{3}$$

$$s.t. \mathbf{v} \in \Psi \tag{4}$$

where f is called the self-paced function controlling the learning, and Ψ denotes the feasible region determined by the curriculum. The self-paced function f determines

how the weights change in $[0, 1]$, and it can be the binary scheme as in (1) or linear, logarithmic can be used [62].

4.2.1.3 Active Self-Paced Learning for Face Recognition

In [1], an active self-paced learning optimization model is proposed. At the beginning a very small number of samples are annotated. When the algorithm runs, a majority is automatically labeled by SPL and a minority is labeled by AL. In this section we give a brief summary of the ASPL formulation with our modifications, and the reader is referred to [1] for further details.

Let training dataset as $D = \{(\mathbf{x}_i, \mathbf{y}_i)\}$, $(i = 1, \ldots, n)$, where x_i is the d-dimensional feature representation of the i^{th} image, and $\mathbf{y}_i$ is the vector representing the ground truth label. It is defined as $\mathbf{y}_i = \{y_i^{(j)} \in \{-1, 1\}\}_{j=1}^{m}$, where $y_i^{(j)}$ corresponds to the label of x_i for the i^{th} subject, as there are m subjects in total. If x_i belongs to the j^{th} subject, then $y_i^{(j)} = 1$, if not $y_i^{(j)} = -1$.

In ASPL, m linear one-vs-rest SVM classifiers are trained using the feature vectors extracted by the CNN. Inspired by [1], we define the function to be minimized as:

$$\min_{\{\mathbf{w},b,\mathbf{v},\mathbf{y}\}} \mathbb{E}(\mathbf{w}, b, \mathbf{y}, \mathbf{v}; \lambda, \Psi) = \sum_{j=1}^{m} \left\| w^{(j)} \right\|_2^2 + C \sum_{i=1}^{n} v_i^{(j)} L(y_i^{(j)}, h(\mathbf{x}_i, \mathbf{w}^{(j)}, b^{(j)})) - f(\mathbf{v}^{(j)}; \lambda_j), \quad (5)$$

$$s.t. \mathbf{v} \in \Psi \quad (6)$$

where $\mathbf{w} = \{w^{(j)}\}_{j=1}^{m} \subset R^d$, $b = \{b^{(j)}\}_{j=1}^{m} \subset R$ are the weight and bias parameters of the m decision functions. C is a regularization parameter between the margin term and the loss term, and it is set to 1 in the experiments. The last term in (6) is the self-paced regularizer for the jth classifier. We use the l_1 norm regularizer in our implementation. The curriculum Ψ is dynamic and determined by the samples selected by the SPL and AL steps, which will be explained below. The loss function $L(.)$ is defined using the hinge loss of $\mathbf{x}_i$ in the j^{th} classifier as $(1 - y_i^j(\mathbf{w}^{(j)T} x_i + b^{(j)}))_+$.

4.2.2 Active Self-Paced Learning with Minimum Sparse Reconstruction

In this section, we give the details of the proposed ASPL-MSR method. We begin with the function in (6), which is minimized using alternative search strategy. The system is initialized by extracting the feature vectors of all labeled and unlabeled samples using the pre-trained CNN. The first step is *classifier updating*, which consists of training m one-vs-rest SVM classifiers, after fixing the current labeled (training) set.

Then, the m SVMs are fixed, and *high-confidence labeling* step is executed first. In [1], linear soft weighting regularizer is used, however in our implementation we used the l_1 norm regularizer for simplicity, which gives us a binary weighting scheme as was explained in (2). Let us denote the distance of x_i to the linear decision boundary of j^{th} classifier as $d_i^{(j)=}(\mathbf{w}^{(j)T}x_i + b^{(j)})/\left\|\mathbf{w}^{(j)}\right\|_2$. Then,

$$v_i = \begin{cases} 1, & \text{if} \max_j d_i^{(j)} > \lambda_{high}, j = 1, \ldots, m \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

which states that the sample is automatically labeled if it is sufficiently far away from the decision boundary.

Next, *low-confidence sample annotating step* (AL) step is executed. An unlabeled sample is considered to be low-confident if the difference between the highest two signed distances to decision boundaries of m classifiers is small (i.e.less than a threshold τ). Instead of manually labeling all low-confidence samples directly, which may have similar samples, we select the most representative (or diverse) ones using minimum sparse reconstruction to reduce the human annotation effort. Mei et al. [92] used MSR to find the most informative (key) frames in videos. We adopt it to select diverse images from the active set in our problem.

Let us call the low-confident samples at the current iteration as the active set and write the feature vectors in a matrix as $F = [f_1,\ f_2 \ldots., f_\alpha] \in \mathbb{R}^{d\times\alpha}$. MSR tries to find the optimal subset $F_K = \left[f_{k_1},\ f_{k_2} \ldots., f_{k_\beta}\right] \in \mathbb{R}^{d\times\beta}$, where $k_1,\ k_2 \ldots., k_\beta \in [1, 2, ..., \alpha]$

so that the original active set F can be accurately reconstructed by F_K and the size of F_K is as small as possible. The objective function to be minimized is:

$$\begin{aligned} \min_S \frac{1}{2} \|F - F_K A\|_2 + \lambda \|S\|_0 \\ s.t. F_K = FS \\ A = f(F, F_K), \end{aligned} \tag{8}$$

where S is a diagonal selection matrix that selects the key images from the active set:

$$S_{ij} = \begin{cases} 0, & i \neq j \\ 0 \text{ or } 1, & i = j \end{cases} \tag{9}$$

and $\|S\|_0$ is the L_0 norm of the selection matrix. The number of nonzero elements in S is β, which is the total number of selected key images. In (8), A represents the reconstruction coefficients of F given the matrix F_K, $f(F, F_K)$ is the reconstruction function, and λ is the weighting coefficient. The first term in (8) minimizes the least-squares reconstruction error (LSRE) and the second term minimizes the number of selected key (diverse) images.

Assuming that β images have been selected and placed in columns of F_K, the next image selected should be the one that gives the maximum LSRE, that the the most dissimilar image to the ones in F_K:

$$f_{k_{\beta+1}} = _{f_j \in F/F_K} \|f_j - F_K a_j\|_2 , \tag{10}$$

where a_j represents the reconstruction coefficients of the jth frame and F/F_K represent the images in the images, which have not been selected. This is equivalent to minimizing:

$$f_{k_{\beta+1}} = _{f_j \in F/F_K} \|F_K a_j\|_2 . \tag{11}$$

Since vectors with large magnitudes tend to have large LSRE, we can normalize with the magnitude and name as the percentage of reconstruction (POR):

$$f_{k_{\beta+1}} =_{f_j \in F/F_K} POR_j =_{f_j \in F/F_K} \frac{\|F_K a_j\|_2}{\|f_j\|_2}. \tag{12}$$

Reconstruction coefficients a_j's are calculated using orthogonal subspace projection (OSP), which projects all the images to the orthogonal space spanned by selected images F_K:

$$a_j = \left(F_K^T F_K\right)^{-1} F_K^T f_j = P_K f_j \tag{13}$$

The algorithm continues to add new images to F_K until the percentage of reconstruction error (POR) is below a pre-determined threshold T_p.

$$POR_j = \frac{\|F_K a_j\|_2}{\|f_j\|_2} < T_p \tag{14}$$

The number of selected key images increases when the threshold T_p is increased. Note that a POR value of 1 means that the jth image can be perfectly reconstructed as a linear combination of the images in F_K. We summarized MSR based image selection algorithm in Algorithm 2.

Algorithm 2: Minimum sparse reconstruction based key (diverse) image selection on active set.

Inputs: Active set $F \in R^{d \times \alpha}$ with α images, where each feature vector is extracted using the fine-tuned CNN.

Output: The key (diverse) image set $F_K \in R^{d \times \beta}$

Initialize the key image set using the first image in the active set $[f_{k_1}]$ and set $\beta = 1$.

Select the second key image as the one with the largest L_2 distance from the first image $[f_{k_1} f_{k_2}]$ and set $\beta = 2$.

Calculate the POR for all images in set F/F_K using (14).

while $POR < T_p$ **do**

  Select the next key image using (12).

  Update key image set $F_K = \left[F_K, f_{k_{\beta+1}}\right]$.

  Calculate the POR for all images in set F/F_K using (14).

  Increase β by one.

end

A toy example for the MSR-based diverse image selection method is shown in Fig. 8. The images in the top row represent the images before selection, which contain similar images. Since MSR tries to reconstruct a dictionary with as few diverse images as possible, similar images are eliminated. In CASIA-WebFace-Sub and CACD datasets, there are indeed similar face images taken with short time differences. Instead of labelling all similar images manually, MSR selects a diverse subset for labelling. Later, unselected similar images are easily automatically labeled by SPL.

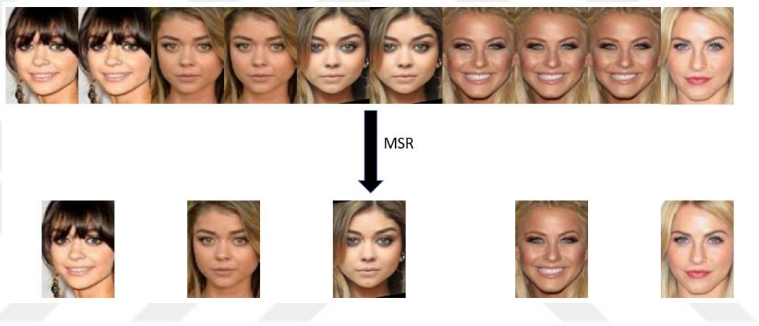

Figure 8: The toy example of the MSR method. The images in the first row represent the candidates for selection, and the images in second row represent the selected images with MSR.

There should be a balance between the number of selected images with self-paced learning and active learning. Labelling low-confidence images with self-paced learning causes mislabelling and reduces accuracy. On the other hand, labelling with active learning requires a lot of effort.

We combine active learning, self-paced learning and minimum sparse reconstruction methods to build a cost-effective framework for face recognition. We take advantage of each approach. SPL selects high-confidence images with no computational cost; AL selects informative images for training, and MSR gives the best representative and diverse set among the informative images to reduce the total number of images that need to be labeled manually.

The main steps for the proposed ASPL-MSR method is given in Algorithm 3.

Algorithm 3: The proposed ASPL-MSR method for face recognition

Inputs: A dataset with %10 annotated ($\{x_i\}$), and %90 not annotated ($\{u_j\}$), Model parameters of the pre-trained CNN (AlexNet).
Output: Optimal model parameters w^*, b^* for m SVM classifiers.
while $\{u\} \neq \emptyset$ **do**
  Use pre-trained CNN to extract feature vectors of images in $\{x_i\}$ and $\{u_j\}$
  for $k = 1 : 3$ **do**
    Train m SVMs using the feature vectors of $\{x_i\}$.
    Find and pseudo-label high-confidence samples using SPL.
    Update $\{x_i\}$ and $\{u_j\}$.
    Train SVMs using the feature vectors of $\{x_i\}$.
    Find and automatically annotate diverse and low-confidence samples using AL-MSR.
    Update $\{x_i\}$ and $\{u_j\}$.
  end
  Fine-tune CNN using $\{x_i\}$.
end
return w^*, b^* for m SVMs.

4.3 Experimental Results

In this section, we give the details of the experimental setup and results using self-paced learning, active learning, active self-paced learning and the proposed active-self-paced learning with minimum sparse reconstruction methods on the two different datasets (CASIA-WebFace-Sub and CACD).

4.3.1 Experimental Setup

We use the 80% of the images for training and 20% of the images for testing. Approximately 10% of the total training data is randomly selected as the initial set and used for fine-tuning the pre-trained CNN (AlexNet) model. The rest of the training data is treated as unlabeled, which will be annotated by the proposed method.

We fine-tune the last 2 fully connected (FC) layers of the pre-trained AlexNet using the initial set. Then, all images are passed through the network and 4096-dimensional feature vectors are obtained from the fc7 layer. We then train the m

linear kernel SVMs with the feature vectors of the initial set. Low-confidence and high-confidence samples are specified based on the distance of each image vector to the one-vs-rest SVM decision boundaries as explained in the previous section. High-confidence images are mostly uninformative since they are far from the linear decision boundaries. Therefore, including them to the training set does not change the accuracy of SVMs significantly. We define the low confidence images based on the distance gap between the two most probable classes. If the distance is below the predefined threshold, the image is selected for the active set.

The thresholds used in SPL, AL and MSR methods determined the number of selected samples from the unlabeled set. The parameter λ_{high} in SPL (7) is selected as 1, which is determined experimentally, and its value does not change during the iterations. Lower λ_{high} values cause more labelling error due to the incorrect annotation, and higher values cause less number of automatically selected images. The threshold for determining low-confidence images in AL is set to 0.0075 initially and it is doubled in each fine-tuning step. The network gets more accurate after each iteration, hence, the number of selected uncertain images decreases. Therefore, we increase the threshold value to keep the total number of selected uncertain images at a certain level. MSR selects the most representative images for training from the active set. Selected images are manually annotated, and unselected images are returned to the unlabeled set. SVMs are retrained after the each SPL and AL-MSR annotation. After three rounds of SPL and AL-MSR updates, the CNN is fine-tuned with the labeled instances. This process continues until all images are labeled. CNN is fine-tuned using the stochastic gradient descent with momentum 0.9, learning rate 0.001, batch size 128 and weight decay 0.0001.

4.3.2 Experimental Results on the Cross-Age Celebrity dataset (CACD)

In order to observe the performance of each component in the proposed method, we report the results of SPL, AL, ASPL and ASPL-MSR in Table 8 and Table 9.

Since the initial set is labeled by human annotation, percentage of automatically labeled samples with SPL begins with 0% in Table 8. According to the results, we observe that SPL does not result in an increase in accuracy after a certain point. Since the selected images are high-confidence, and far from the decision boundary, they have almost no impact on SVM performance. Moreover, the rate of increase in the number of images joining the training set decreases after each iteration. AL tends to select informative samples and the increase in accuracy is faster than using random selection. Accuracy almost reached the same level (92.71%) as training with whole dataset when the total number of annotated samples are 45.93% of the training set.

When ASPL is used (Table 9), the same accuracy (92.79%) was achieved when 40.15% of the entire dataset is annotated manually, which is almost 6% lower than the percentage of total manually annotated images using AL method. In the ASPL method when network gets more accurate with training low-confidence samples, high-confidence samples are labeled automatically, which are helpful for fine-tuning the CNN and increases the accuracy eventually.

If we analyze the results of the proposed ASPL-MSR method (Table 9), we observe that an accuracy of 92.81% is obtained when 36.90% of the data is human annotated, which indicates 3.25% reduction in manually annotated samples as compared to ASPL. It proves that MSR selects the best possible diverse subset to represent images selected with active learning. Therefore, we obtained a more accurate network with less data (less is more).

Table 8: Percentage of labeled samples and corresponding accuracies using SPL (left) and AL (right) methods on CACD dataset.

Percentage of Automatically Labeled Samples (%)	Accuracy (%)
0	49.08
1.51	51.57
2.99	51.64
3.87	51.62
4.77	51.36
5.44	51.78
6.12	51.59

Percentage of Manually Labeled Samples (%)	Accuracy (%)
9.48	49.08
18.87	73.67
27.42	83.50
34.18	88.01
38.52	89.93
41.58	90.44
45.93	92.71

Table 9: Percentage of manually labeled samples and corresponding accuracies for ASPL (left) and the proposed ASPL-MSR (right) methods on CACD dataset.

Percentage of Manually Labeled Samples (%)	Accuracy (%)
9.48	49.08
15.94	66.60
22.81	81.21
28.84	86.78
33.18	89.66
35.79	90.50
38.03	92.00
40.15	92.79

Percentage of Manually Labeled Samples (%)	Accuracy (%)
9.48	49.58
14.31	65.50
19.99	77.64
25.97	84.83
31.10	89.21
34.52	90.97
36.90	92.81

We also present the experimental results on CACD dataset using plots for better analysis. In Fig. 9, we can see the plots of accuracies of SPL, AL, ASPL and the proposed ASPL-MSR methods versus the percentage of manually annotated samples. The dashed black line is the maximum accuracy (92.87%) achieved when we use the whole CACD dataset using the ground truth labels. Since the total number of manually annotated samples does not change during training with SPL, its maximum accuracy (51.78%) is shown with a flat blue line as selecting only high-confidence samples limits the accuracy that can be achieved. AL (purple line) and ASPL (green line) reached the maximum accuracy with less percentage of labeled images, 45.93% and 40.15%, respectively. The proposed ASPL-MSR method achieves the maximum accuracy using the least number of human annotated data (36.90%).

In Fig. 10 we give the plots of accuracies versus the iteration number. One iteration includes a fine-tuning of the CNN. We observed a slight increase in accuracy when using SPL approach. On the other hand ASPL-MSR approach achieved more than 90% accuracy at the end of last iteration.

Percentage of manually labeled samples versus the number of iterations plots can be seen in Fig. 11. AL and ASPL with MSR methods reached the maximum accuracy at the end of seventh iteration. However, ASPL lasted eight iterations, since we don't restrict the number of iterations in our implemention. Experiments are stopped when the accuracy reaches the same level as training with all images using ground truth labels. According to the graph in Fig. 11, the ASPL-MSR method requires the least number of hand labeling.

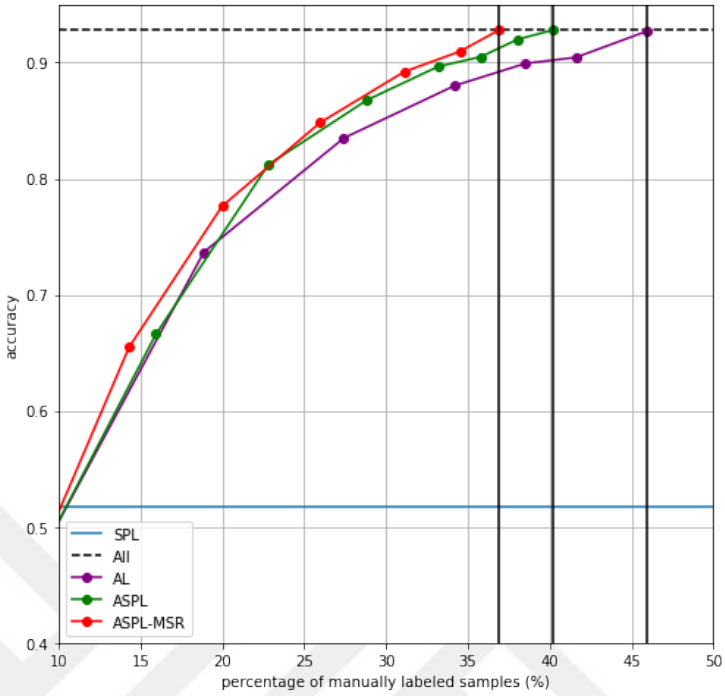

Figure 9: Accuracies of the 4 different methods (SPL, AL, ASPL, proposed ASPL-MSR) versus the percentage of manually labeled samples on the CACD dataset.

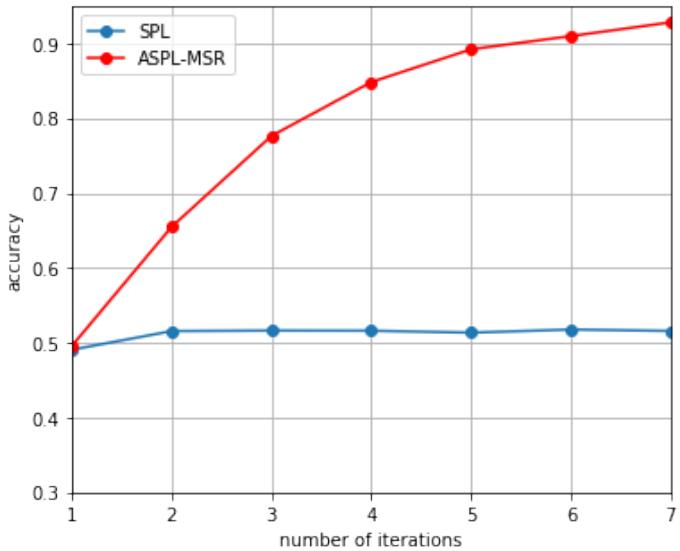

Figure 10: Accuracies of SPL and proposed ASPL-MSR versus the number of iterations on the CACD dataset.

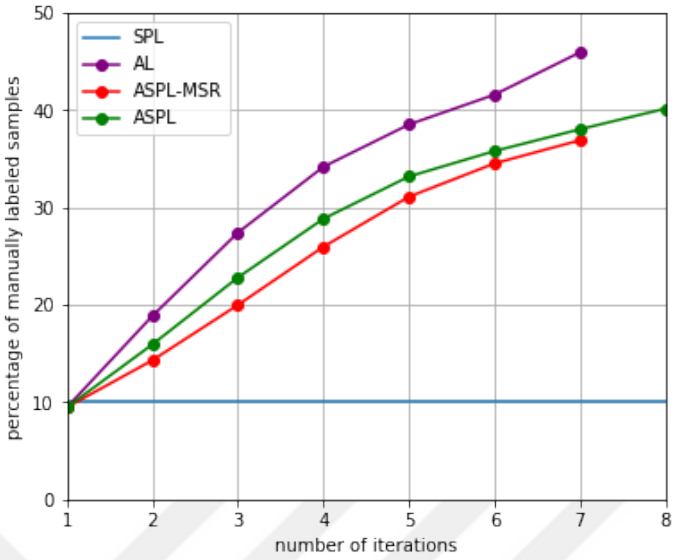

Figure 11: Accuracy of the 4 different methods with the increase of number of iterations on the CACD dataset.

In Table 10 and Table 11, we give the details about each iteration for ASPL and ASPL-MSR methods, respectively. The total number of selected images with SPL and AL in each fine-tuning, and number of disagreeing annotations for samples labeled automatically are shown. There are 3 SPL and AL updates between each fine-tuning. We can observe that the number of selected images with SPL increases with each fine-tuning initially. As the training set increases, the network becomes more accurate, so the number of high-confidence samples increases. The reason for the drop in last iterations is the decrease in total number of training images. Since we double the threshold for AL at each iteration, the number of images selected by AL do not drop too much. The number of total disagreements between the ground truth and automatically assigned labels is 90 in Table 11. 49 of them correspond to the incorrect ground truth labels corrected by SPL, and 33 labels were annotated incorrectly. It was not decided whether 8 of the labels were annotated correctly or not since the images were ambigous.

In Table 12, we give a summary and comparison of performance of different methods on CACD dataset. We can see that our ASPL-MSR method requires less manually annotated training samples than state-of-the-art methods.

Table 10: Iteration details for ASPL method on CACD dataset. Total number of selected images with SPL and AL in each fine-tuning, and number of disagreeing annotations for samples labeled automatically are shown.

Fine-tuning	AL Updates	Total Selected Images	Selected with SPL	Number of Disagreeing Annotations	Selected with AL
Fine-tuning 1	Update 1	1807	650	3	1157
	Update 2	1228	310	1	918
	Update 3	1111	304	3	807
Fine-tuning 2	Update 1	2799	1569	5	1230
	Update 2	1662	641	5	1021
	Update 3	1360	539	10	821
Fine-tuning 3	Update 1	4313	3158	21	1155
	Update 2	1625	766	3	859
	Update 3	1262	583	3	679
Fine-tuning 4	Update 1	4633	3658	15	975
	Update 2	1329	738	8	591
	Update 3	796	425	4	371
Fine-tuning 5	Update 1	3491	2711	10	780
	Update 2	760	476	1	284
	Update 3	312	206	0	106
Fine-tuning 6	Update 1	2838	2041	6	797
	Update 2	527	357	1	170
	Update 3	150	116	0	34
Fine-tuning 7	Update 1	2083	1279	7	804
	Update 2	420	301	0	119
	Update 3	107	84	0	23

Table 11: Iteration details for proposed ASPL-MSR method on CACD dataset. Total number of selected images with SPL and AL in each fine-tuning, and number of disagreeing annotations for samples labeled automatically are shown.

Fine-tuning	AL Updates	Total Selected Images	Selected with SPL	Number of Disagreeing Annotations	Selected with AL-MSR
Fine-tuning 1	Update 1	1397	650	3	747
	Update 2	972	246	0	726
	Update 3	914	229	0	685
Fine-tuning 2	Update 1	2538	1618	8	920
	Update 2	1272	439	3	833
	Update 3	1202	420	3	782
Fine-tuning 3	Update 1	3432	2428	13	1004
	Update 2	1547	666	3	881
	Update 3	1358	572	3	786
Fine-tuning 4	Update 1	4324	3367	13	957
	Update 2	1434	674	6	760
	Update 3	1111	537	3	574
Fine-tuning 5	Update 1	3814	2969	16	845
	Update 2	1028	559	0	469
	Update 3	531	316	2	215
Fine-tuning 6	Update 1	2970	2214	9	756
	Update 2	687	448	5	239
	Update 3	225	158	0	67

Table 12: Summary and comparison of the different methods on CACD dataset. The percentage of manually annotated data to achieve the maximum accuracy is given.

Method	Total Manually Annotated Samples (%)
AL	45.93
ASPL	40.15
ASPL-MSR (Ours)	**36.90**
Lin [1]	41.00

4.3.3 Experimental Results on the CASIA-WebFace-Sub dataset

In order to observe the performance of each component in the proposed method, we report the results of SPL, AL, ASPL and ASPL-MSR in Table 13 and Table 14. The maximum accuracy that can be achieved when we use all of the training data with ground truth labels is 81.72%.

In Table 13, since the initial set is labeled by human annotation, percentage of automatically labeled samples begins with a value of 0%. We observe that the accuracy of SPL does not increase after a certain point. The accuracy of AL almost reached the maximum level (81.52%) using manual labeling for 64.07% of the training set.

In Table 14, we observe that the maximum accuracy was achieved when 55.83% of the entire dataset is annotated manually using ASPL, which is almost 10% lower than the percentage of total manually annotated images using AL method. Using the proposed ASPL-MSR method the maximum accuracy of 81.59% is achieved using manual annotation of 54.10% of training data. This indicates 1.73% improvement on reducing the percentage of manually annotated samples as compared to ASPL approach. Therefore, we obtained a more accurate network with less annotated data.

Table 13: Percentage of labeled samples and corresponding accuracies using SPL (left) and AL (right) methods on CASIA-WebFace-Sub dataset.

Percentage of Automatically Labeled Samples (%)	Accuracy (%)
0	33.07
0.94	37.46
1.61	38.32
2.15	38.43
2.53	39.20
2.92	39.41
3.25	40.13
3.56	40.11
3.82	40.08

Percentage of Manually Labeled Samples (%)	Accuracy (%)
9.97	33.07
18.65	52.93
29.16	66.80
39.23	73.15
47.47	77.05
52.77	78.40
56.15	79.57
59.30	80.25
64.07	81.52

Table 14: Percentage of manually labeled samples and corresponding accuracies for ASPL (left) and the proposed ASPL-MSR (right) methods on CASIA-WebFace-Sub dataset.

Percentage of Manually Labeled Samples (%)	Accuracy (%)
9.97	33.07
19.24	55.75
28.89	67.04
38.46	73.69
45.99	77.37
50.69	79.36
53.43	80.11
55.83	81.55

Percentage of Manually Labeled Samples (%)	Accuracy (%)
9.97	33.07
16.72	51.05
23.58	61.99
30.60	69.27
37.62	73.93
44.62	77.41
51.14	79.84
54.10	81.59

The plots of experimental results on the CASIA-WebFace-Sub dataset are given in Fig. 13, Fig. 14, and Fig. 15. The ASPL-MSR is shown with the red curve in all figures, which indicate that it requires the least number of human annotated data to achieve the maximum accuracy.

In Table 15 and Table 16, we give the details about each iteration for ASPL and ASPL-MSR methods, respectively. The total number of selected images with SPL and AL in each fine-tuning, and number of disagreeing annotations for samples labeled automatically are shown. There are 3 SPL and AL updates between each fine-tuning. If we compare the last columns of both tables, we can see that the number of images selected by AL-MSR is less than the number of images selected by AL at each update and iteration (update). We can also observe that the number of selected images with

SPL increases in each fine-tuning initially. As the training set increases, the network becomes more accurate, so the number of high-confidence samples increases. The reason for the drop in last iterations is the decrease in the total number of unlabeled images. The number of disagreements between the ground truth and pseudo-labels is 145, 86 of which belongs to corrected ground truth labels. However, 48 labels were annotated incorrectly by ASPL-MSR, and 11 images were ambiguous. Several examples of images with incorrect ground truth labels are shown in Fig. 12.

Figure 12: Examples of the images with incorrect ground truth labels on CASIA-WebFace-Sub dataset. Folder locations of the images in the first row are 0719637/296.jpg, 2650334/095.jpg, 0781981/015.jpg, 0301959/111.jpg, 0000532/095.jpg, 0000439/471.jpg respectively. Folder locations of the images in the second row are 0186505/148.jpg, 0339011/052.jpg, 0000233/223.jpg, 0074477/072.jpg, 0272479/031.jpg, 1433588/237.jpg respectively.

In Table 17, we give a summary and comparison of performance of different methods on CASIA-WebFace-Sub dataset. We can see that our ASPL-MSR method requires less manually annotated training samples than state-of-the-art methods.

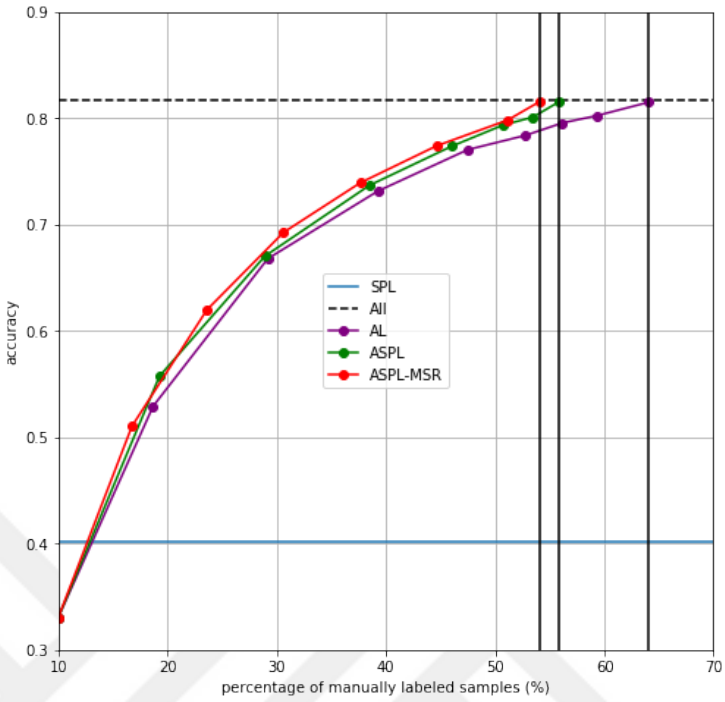

Figure 13: Accuracies of the 4 different methods (SPL, AL, ASPL, proposed ASPL-MSR) versus the percentage of manually labeled samples on the CASIA-WebFace-Sub dataset.

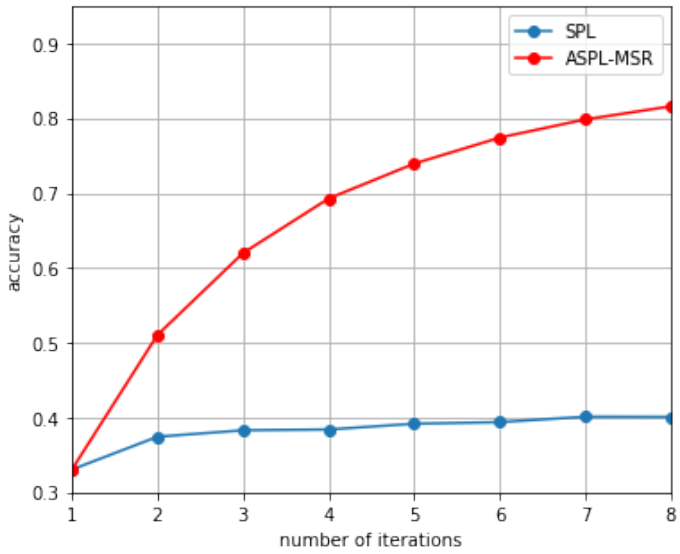

Figure 14: Accuracies of SPL and proposed ASPL-MSR versus the number of iterations on the CASIA-WebFace-Sub dataset.

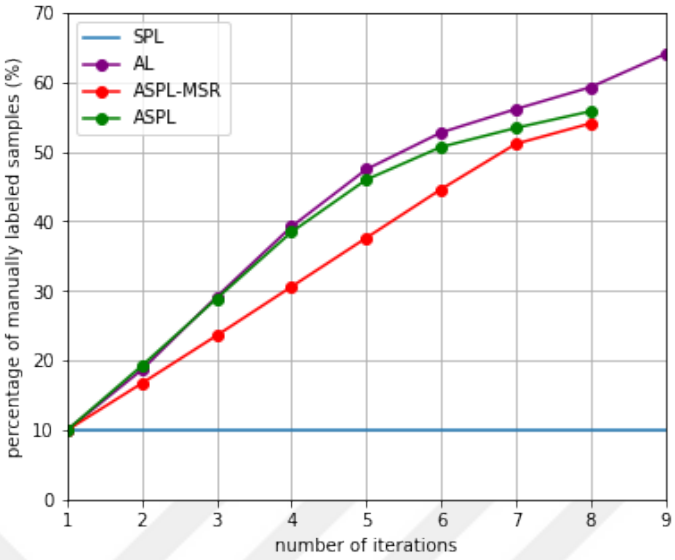

Figure 15: Accuracy of the 4 different methods with the increase of number of iterations on the CASIA-WebFace-Sub dataset.

Table 15: Iteration details for ASPL method on CASIA-WebFace-Sub dataset. Total number of selected images with SPL and AL in each fine-tuning, and number of disagreeing annotations for samples labeled automatically are shown.

Fine-tuning	AL/SPL Updates	Total Selected Images	Selected with SPL	Number of Disagreeing Annotations	Selected with AL
Fine-tuning 1	Update 1	6718	1360	3	5358
	Update 2	5271	1042	3	4229
	Update 3	4701	850	6	3851
Fine-tuning 2	Update 1	10.530	4925	4	5605
	Update 2	6318	1738	4	4580
	Update 3	5420	1594	2	3826
Fine-tuning 3	Update 1	14.303	8406	15	5897
	Update 2	6779	2320	11	4459
	Update 3	5247	1732	7	3515
Fine-tuning 4	Update 1	13.229	7870	16	5359
	Update 2	5320	1949	6	3371
	Update 3	3434	1234	5	2200
Fine-tuning 5	Update 1	10.173	6003	22	4170
	Update 2	3229	1364	5	1865
	Update 3	1437	651	4	786
Fine-tuning 6	Update 1	7500	4364	13	3136
	Update 2	1507	831	5	676
	Update 3	387	226	4	161
Fine-tuning 7	Update 1	5494	2463	15	3031
	Update 2	1034	653	3	381
	Update 3	179	110	0	69

Table 16: Iteration details for proposed ASPL-MSR method on CASIA-WebFace-Sub dataset. Total number of selected images with SPL and AL in each fine-tuning, and number of disagreeing annotations for samples labeled automatically are shown.

Fine-tuning	AL/SPL Updates	Total Selected Images	Selected with SPL	Number of Disagreeing Annotations	Selected with AL-MSR
Fine-tuning 1	Update 1	4659	1361	3	3298
	Update 1	3916	647	1	3269
	Update 1	3884	653	4	3231
Fine-tuning 2	Update 1	6488	3143	4	3345
	Update 2	4253	931	4	3322
	Update 3	4197	907	3	3290
Fine-tuning 3	Update 1	9323	5903	9	3420
	Update 2	4776	1384	1	3392
	Update 3	4580	1213	3	3367
Fine-tuning 4	Update 1	10682	7252	13	3430
	Update 2	4807	1411	7	3396
	Update 3	4754	1399	5	3355
Fine-tuning 5	Update 1	10537	7087	22	3450
	Update 2	4855	1464	3	3391
	Update 3	4533	1223	6	3310
Fine-tuning 6	Update 1	9492	6059	16	3433
	Update 2	4567	1254	5	3313
	Update 3	3769	1050	8	2719
Fine-tuning 7	Update 1	5918	4281	19	1637
	Update 2	2002	563	5	1439
	Update 3	1647	440	4	1207

Table 17: Summary and comparison of the different methods on CASIA-WebFace-Sub dataset. The percentage of manually annotated data to achieve the maximum accuracy is given.

Method	Total Annotated Samples (%)
AL	64.07
ASPL	55.83
ASPL-MSR (Ours)	**54.10**
Lin [1]	56.00

4.4 Summary

In this chapter, we have presented a self-annotating framework to reduce the human annotation effort and correct noisy labels. According to the results, when 36.90% and 54.10% of the training images were manually annotated in the CACD and CASIA-WebFace-Sub datasets, respectively, the same face recognition accuracy was achieved as compared to using 100% of the ground truth labels.

CHAPTER V

CONCLUSION AND FUTURE WORK

We presented two novel methods to improve the performance of face recognition systems. In the first method, the training image set is introduced to the CNN based on their complexity levels, which is also known as curriculum learning. These complexity levels are determined based on the head pose angles of the faces. Thereby, the network converged to a better local minimum, and the face recognition accuracy increased significantly. According to the results on CASIA-WebFace-Sub dataset, the accuracy increased by 1.47% when the training set was divided into four subsets of increasing difficulty instead of training with random ordering. As for future work, attributes other than the head pose can be used to assess the difficulty of face images since frontal images can be noisy or blurry. In such cases, images that contain blur and noise could be considered as difficult images.

In the second method, we have presented an incremental face recognition framework to reduce human annotation effort while keeping the maximum possible accuracy, and correct noisy labels. We proposed the utilization of minimum-sparse-reconstruction based diverse subset selection for the low-confidence samples. Experimental results on the large scale CASIA-WebFace-Sub and CACD datasets indicate that number of human annotated samples is reduced significantly while obtaining comparable accuracy to the case when the whole training data is used with ground truth labels. To the best of our knowledge, this is the first work, which uses MSR to represent selected images with active learning using a dictionary reflecting diversity, which is simpler to implement. As for future work, our approach could also be applied on video datasets.

APPENDIX A

CONVOLUTIONAL NEURAL NETWORKS

Convolutional neural networks (CNN) improved the state-of-the-art in recognition tasks [93]. In this thesis, we used CNN to extract feature vectors and classification. We have summarized CNN for a better understanding of the thesis.

Artificial Neural Network is inspired by the working principle of the humans' biological neural systems. There are billions of the neurons in the human brain, and these are connected to each other with dendrites. Dendrites carry output of the neurons to the input of the other neurons. Therefore, information is carried between neurons. The computational unit of the ANNs is also called neuron. Weighted sum of the each input is transferred to the next neuron after adds up with the bias. Hence, the score value of the each neuron in the output layer is calculated. To measure the effectiveness of the network, we formulate loss function. Then, we try to minimize it while updating the parameters of neurons. There are different loss functions. In the thesis, we used cross-entropy loss:

$$L_i = -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right),$$

where f_{y_i} is the class score of the correct class and f_j is the class score of the jth class. The parameters of the neurons are updated based on the loss function. Backpropagation is an algorithm of taking the gradient of the loss function with respect to the weights. Stochastic gradient descent was used to update parameters in an iterative way. It aims to find minimum loss value. Batch of samples are selected randomly, and the new parameters are calculated as $newparameter = oldparameter - (gradient * learningrate)$, where the learning rate is a hyperparameter

that controls how much the model will converge.

CNNs are the specialized ANNs, which we use to classify face images. It takes image as an input and differentiates each one of them from the other classes. CNNs consist of input layer, convolutional layer, pooling layer and fully connected layer. Also, non-linear activation functions are applied to output of the neurons. Activation functions decide whether the neurons will be activated or not. We fine-tuned the CASIA-WebFace-Sub dataset with AlexNet; therefore, input dimension is converted to pre-defined size, which is 224x224x3. Convolutional layer applies the filter to the input to extract feature map. Pooling layer performs a downsampling and reduces the size of the convolved feature. Lastly, fully connected layers compute the score value of each class.

APPENDIX B

ONE-VS-REST SUPPORT VECTOR MACHINES

In this thesis, support vector machine (SVM) was used to specify the confidence of the unlabeled images. We have summarized SVM for a better understanding of the thesis.

SVM is a supervised machine learning algorithm, which is mainly used for classification and regression. The main purpose of the SVM is to find hyperplanes that separate the classes between each one of them. We used special type of SVM, which is one-vs-rest SVM. It consists one binary classifier per class. The one-vs-rest SVM converts multi-class classification into one binary classification for each class. That's why, the number of hyperplanes equals the total number of classes. If the label of the data point x_i belongs to the correct class, its label is determined as 1, if not, it is -1. Hence, classes of the data points are specified based on the data point is belonged whether $wx + b > 1$ or $wx + b < -1$, where $wx + b = 0$ is the hyperplane that separates classes. We can formulate our SVM approach for binary classification as:

$$f(x) = \begin{cases} wx_i + b \geq 1, & \text{if } y_i = +1 \\ wx_i + b \leq 1, & \text{if } y_i = -1 \end{cases}$$

We can generalise this formula as:

$$f(x) = \Big\{ y_i(wx_i + b) \geq 1, \quad \text{for all } i$$

According to the formula, we try to maximize the margin $M = \frac{2}{|w|}$, which is the same as minimizing the $\frac{1}{2}w^t w$, where $y_i(wx_i + b) \geq 1$ is the constraint.

Bibliography

[1] L. Lin, K. Wang, D. Meng, W. Zuo, and L. Zhang, "Active self-paced learning for cost-effective and progressive face identification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, pp. 7–19, 2018.

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

[3] T. Kanade, *Computer Recognition of Human Faces*. Birkhauser Verlag, 1977.

[4] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM Computing Surveys*, vol. 35, pp. 399–458, 2003.

[5] A. F. Abate, M. Nappi, D. Riccio, and G. Sabatino, "2d and 3d face recognition: A survey," *Pattern Recognition Letters*, vol. 28, pp. 1885–1906, 2007.

[6] M. Taskiran, N. Kahraman, and C. E. Erdem, "Face recognition: Past, present and future (a review)," *Digital Signal Processing*, vol. 106, p. 102809, 2020.

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

[8] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," 2015.

[9] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 815–823, 2015.

[10] R. Ranjan, C. D. Castillo, and R. Chellappa, "L2-constrained softmax loss for discriminative face verification," *arXiv preprint arXiv:1703.09507*, 2017.

[11] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in *IEEE Conf. on Automatic Face and Gesture Recognition*, 2018.

[12] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 4690–4699, 2019.

[13] C. Huang, Y. Li, C. L. Chen, and X. Tang, "Deep imbalanced learning for face recognition and attribute prediction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

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

[15] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," *Image and Vision Computing*, vol. 28, pp. 807–813, 2010. Best of Automatic Face and Gesture Recognition 2008.

[16] I. Kemelmacher-Shlizerman, S. M. Seitz, D. Miller, and E. Brossard, "The megaface benchmark: 1 million faces for recognition at scale," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 4873–4882, 2016.

[17] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "Ms-celeb-1m: A dataset and benchmark for large-scale face recognition," in *European Conf. on Computer Vision*, pp. 87–102, Springer, 2016.

[18] A. Bansal, A. Nanduri, C. D. Castillo, R. Ranjan, and R. Chellappa, "Umdfaces: An annotated face dataset for training deep networks," in *IEEE Int. Joint Conf. on Biometrics*, pp. 464–473, IEEE, 2017.

[19] B. Maze, J. Adams, J. A. Duncan, N. Kalka, T. Miller, C. Otto, A. K. Jain, W. T. Niggel, J. Anderson, J. Cheney, *et al.*, "Iarpa janus benchmark-c: Face dataset and protocol," in *Int. Conf. on Biometrics*, pp. 158–165, IEEE, 2018.

[20] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in *IEEE Int. Conf. on Automatic Face & Gesture Recognition*, pp. 67–74, IEEE, 2018.

[21] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep deatures for scene recognition using places database," in *Advances in Neural Information Processing Systems*, pp. 487–495, 2014.

[22] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Int. Conf. on Machine Learning*, 2009.

[23] A. Pentina, V. Sharmanska, and C. H. Lampert, "Curriculum learning of multiple tasks," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 5492–5500, 2015.

[24] V. Avramova, "Curriculum learning with deep convolutional neural networks," Master's thesis, KTH Royal Institute of Technology, 2015.

[25] L. Gui, T. Baltrusaitis, and L. P. Morency, "Curriculum learning for facial expression recognition," in *IEEE Int. Conf. on Automatic Face and Gesture Recognition*, pp. 505–511, 2017.

[26] M. Wang and W. Deng, "Deep face recognition: A survey," *CoRR*, vol. abs/1804.06655, 2018.

[27] R. Ranjan, S. Sankaranarayanan, A. Bansal, N. Bodla, J. C. Chen, V. M. Patel, C. D. Castillo, and R. Chellappa, "Deep learning for understanding faces: Machines may be just as good, or better, than humans," *IEEE Signal Processing Magazine*, vol. 35, pp. 66–83, 2018.

[28] G. Guo and N. Zhang, "A survey on deep learning based face recognition," *Computer Vision and Image Understanding*, vol. 189, p. 102805, 2019.

[29] M. P. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," in *Advances in Neural Information Processing Systems*, pp. 1189–1197, 2010.

[30] Y. J. Lee and K. Grauman, "Learning the easy things first: Self-paced visual category discovery," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1721–1728, 2011.

[31] J. S. Supancic and D. Ramanan, "Self-paced learning for long-term tracking," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2379–2386, 2013.

[32] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *SIGIR'94*, pp. 3–12, Springer, 1994.

[33] T. Scheffer, C. Decomain, and S. Wrobel, "Active hidden markov models for information extraction," in *Int. Symposium on Intelligent Data Analysis*, pp. 309–318, Springer, 2001.

[34] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *Journal of machine learning research*, vol. 2, pp. 45–66, 2001.

[35] B.-C. Chen, C.-S. Chen, and W. H. Hsu, "Cross-age reference coding for age-invariant face recognition and retrieval," in *European Conf. on Computer Vision*, pp. 768–783, Springer, 2014.

[36] B. Buyuktas, C. Eroglu Erdem, and A. T. Erdem, "Curriculum learning for face recognition," in *European Conf. on Signal Processing*, 2020.

[37] D. G. Lowe, "Object recognition from local scale-invariant features.," in *Int. Conf. Computer Vision*, vol. 99, pp. 1150–1157, 1999.

[38] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European Conf. on Computer Vision*, pp. 404–417, Springer, 2006.

[39] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 1, pp. 886–893, 2005.

[40] H. Fan, Z. Cao, Y. Jiang, Q. Yin, and C. Doudou, "Learning deep face representation," *arXiv preprint arXiv:1403.2802*, 2014.

[41] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *European Conf. on Computer Vision*, pp. 499–515, Springer, 2016.

[42] B. F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, and A. K. Jain, "Pushing the frontiers of unconstrained face detection and recognition: Iarpa janus benchmark a," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1931–1939, 2015.

[43] I. Masi, F.-J. Chang, J. Choi, S. Harel, J. Kim, K. Kim, J. Leksut, S. Rawls, Y. Wu, T. Hassner, *et al.*, "Learning pose-aware models for pose-invariant face recognition in the wild," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, pp. 379–393, 2018.

[44] S. Ge, S. Zhao, C. Li, and J. Li, "Low-resolution face recognition in the wild via selective knowledge distillation," *IEEE Transactions on Image Processing*, vol. 28, pp. 2051–2062, 2018.

[45] E. Zangeneh, M. Rahmati, and Y. Mohsenzadeh, "Low resolution face recognition using a two-branch deep convolutional neural network architecture," *Expert Systems with Applications*, vol. 139, p. 112854, 2020.

[46] H. Gao, H. K. Ekenel, and R. Stiefelhagen, "Pose normalization for local appearance-based face recognition," in *Int. Conf. on Biometrics*, pp. 32–41, Springer, 2009.

[47] C. Ding and D. Tao, "Trunk-branch ensemble convolutional neural networks for video-based face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, pp. 1002–1014, 2017.

[48] W. AbdAlmageed, Y. Wu, S. Rawls, S. Harel, T. Hassner, I. Masi, J. Choi, J. Lekust, J. Kim, P. Natarajan, *et al.*, "Face recognition using deep multi-pose representations," in *IEEE Winter Conf. on Applications of Computer Vision*, pp. 1–9, IEEE, 2016.

[49] M. Karasuyama and I. Takeuchi, "Multiple incremental decremental learning of support vector machines," *IEEE Transactions on Neural Networks*, vol. 21, pp. 1048–1059, 2010.

[50] S. Ozawa, S. L. Toh, S. Abe, S. Pang, and N. Kasabov, "Incremental learning of feature space and classifier for face recognition," *Neural Networks*, vol. 18, pp. 575–584, 2005.

[51] A. V. Kelly, *The Curriculum: Theory and Practice*. Sage, 2009.

[52] J. L. Elman, "Learning and development in neural networks: The importance of starting small," *Cognition*, vol. 48, pp. 71–99, 1993.

[53] V. I. Spitkovsky, H. Alshawi, and D. Jurafsky, "From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing," in *Human Language Technologies: The 2010 Annual Conf. of the North American Chapter of the Association for Computational Linguistics*, pp. 751–759, 2010.

[54] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Int. Conf. on Machine Learning*, pp. 41–48, ACM, 2009.

[55] A. Lapedriza, H. Pirsiavash, Z. Bylinskii, and A. Torralba, "Are all training examples equally valuable?," *arXiv preprint arXiv:1311.6510*, 2013.

[56] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," in *Advances in Neural Information Processing Systems*, pp. 1171–1179, 2015.

[57] L. Gui, T. Baltrušaitis, and L.-P. Morency, "Curriculum learning for facial expression recognition," in *IEEE Int. Conf. on Automatic Face & Gesture Recognition*, pp. 505–511, IEEE, 2017.

[58] L. Jiang, D. Meng, S.-I. Yu, Z. Lan, S. Shan, and A. Hauptmann, "Self-paced learning with diversity," in *Advances in Neural Information Processing Systems*, pp. 2078–2086, 2014.

[59] Y. Tang, Y.-B. Yang, and Y. Gao, "Self-paced dictionary learning for image classification," in *ACM Int. Conf. on Multimedia*, pp. 833–836, 2012.

[60] L. Jiang, D. Meng, T. Mitamura, and A. G. Hauptmann, "Easy samples first: Self-paced reranking for zero-example multimedia search," in *ACM Int. Conf. on Multimedia*, pp. 547–556, 2014.

[61] Q. Zhao, D. Meng, L. Jiang, Q. Xie, Z. Xu, and A. G. Hauptmann, "Self-paced learning for matrix factorization," in *AAAI Conf. on Artificial Intelligence*, 2015.

[62] L. Jiang, D. Meng, Q. Zhao, S. Shan, and A. G. Hauptmann, "Self-paced curriculum learning," in *AAAI Conf. on Artificial Intelligence*, 2015.

[63] H. Li, M. Gong, D. Meng, and Q. Miao, "Multi-objective self-paced learning," in *AAAI Conf. on Artificial Intelligence*, 2016.

[64] S. Harada, J. Lester, K. Patel, T. S. Saponas, J. Fogarty, J. A. Landay, and J. O. Wobbrock, "Voicelabel: using speech to label mobile sensor data," in *Int. Conf. on Multimodal interfaces*, pp. 69–76, 2008.

[65] X. Zhu, J. Lafferty, and R. Rosenfeld, "Semi-supervised learning with graphs (ph. d. thesis)," *Pittsburgh, PA, USA*, 2005.

[66] K. Meshgi and S. Oba, "Active collaboration of classifiers for visual tracking," *Human-Robot Interaction: Theory and Application*, p. 101, 2018.

[67] D. D. Lewis and J. Catlett, "Heterogeneous uncertainty sampling for supervised learning," in *Machine learning proceedings 1994*, pp. 148–156, Elsevier, 1994.

[68] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Fifth annual workshop on Computational learning theory*, pp. 287–294, 1992.

[69] B. Settles, M. Craven, and S. Ray, "Multiple-instance active learning," in *Advances in neural information processing systems*, pp. 1289–1296, 2008.

[70] N. Roy and A. McCallum, "Toward optimal active learning through monte carlo estimation of error reduction," *ICML, Williamstown*, pp. 441–448, 2001.

[71] B. Settles and M. Craven, "An analysis of active learning strategies for sequence labeling tasks," in *Conf. on Empirical Methods in Natural Language Processing*, pp. 1070–1079, 2008.

[72] E. Y. Chang, S. Tong, K. Goh, and C.-W. Chang, "Support vector machine concept-dependent active learning for image retrieval," *IEEE Transactions on Multimedia*, vol. 2, pp. 1–35, 2005.

[73] A. Culotta and A. McCallum, "Reducing labeling effort for structured prediction tasks," in *AAAI Conf. on Artificial Intelligence*, vol. 5, pp. 746–751, 2005.

[74] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell, "Active learning with gaussian processes for object categorization," in *IEEE Int. Conf. on Computer Vision*, pp. 1–8, IEEE, 2007.

[75] A. Holub, P. Perona, and M. C. Burl, "Entropy-based active learning for object recognition," in *IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, pp. 1–8, IEEE, 2008.

[76] A. J. Joshi, F. Porikli, and N. Papanikolopoulos, "Multi-class active learning for image classification," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2372–2379, IEEE, 2009.

[77] A. J. Joshi, F. Porikli, and N. P. Papanikolopoulos, "Scalable active learning for multiclass image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, pp. 2259–2273, 2012.

[78] L. Lin, K. Wang, D. Meng, W. Zuo, and L. Zhang, "Active self-paced learning for cost-effective and progressive face identification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, pp. 7–19, 2017.

[79] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," 2008.

[80] A. Dutta, R. Veldhuis, and L. Spreeuwers, "The impact of image quality on the performance of face recognition," in *33rd WIC Symposium on Information Theory in the Benelux*, pp. 141–148, 2012.

[81] L.-P. Morency, J. Whitehill, and J. Movellan, "Generalized adaptive view-based appearance model: Integrated framework for monocular head pose estimation," in *IEEE Int. Conf. on Automatic Face & Gesture Recognition*, pp. 1–8, IEEE, 2008.

[82] X. Liu, H. Lu, and H. Luo, "A new representation method of head images for head pose estimation," in *IEEE Int. Conf. on Image Processing*, pp. 3585–3588, IEEE, 2009.

[83] G. Fanelli, J. Gall, and L. Van Gool, "Real time head pose estimation with random regression forests," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 617–624, IEEE, 2011.

[84] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 7291–7299, 2017.

[85] T. Baltrušaitis, P. Robinson, and L.-P. Morency, "Openface: An open source facial behavior analysis toolkit," in *IEEE Winter Conf. on Applications of Computer Vision*, pp. 1–10, IEEE, 2016.

[86] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L.-P. Morency, "Openface 2.0: Facial behavior analysis toolkit," in *IEEE Int. Conf. on Automatic Face & Gesture Recognition*, pp. 59–66, 2018.

[87] G. Hacohen and D. Weinshall, "On the power of curriculum learning in training deep networks," in *Int. Conf. on Machine Learning*, 2019.

[88] L. Wiskott, J.-M. Fellous, N. Krüger, and C. Von Der Malsburg, "Face recognition by elastic bunch graph matching," in *Int. Conf. on Computer Analysis of Images and Patterns*, pp. 456–463, Springer, 1997.

[89] E. Alpaydin, *Introduction to Machine Learning*. MIT press, 2009.

[90] Z. Lei, D. Yi, and S. Z. Li, "Learning stacked image descriptor for face recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, pp. 1685–1696, 2016.

[91] J. Gorski, F. Pfeuffer, and K. Klamroth, "Biconvex sets and optimization with biconvex functions: a survey and extensions," *Math Meth Oper Res*, vol. 66, pp. 373—-407, 2007.

[92] S. Mei, G. Guan, Z. Wang, S. Wan, M. He, and D. D. Feng, "Video summarization via minimum sparse reconstruction," *Pattern Recognition*, vol. 48, pp. 522–533, 2015.

[93] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, p. 436, 2015.