2020

Deep Siamese Neural Networks for Facial Expression Recognition in the Wild

Wassan Hayale
University of Denver

Deep Siamese Neural Networks for Facial Expression Recognition in the Wild

---

A Dissertation

Presented to

the Faculty of the Daniel Felix Ritchie School of Engineering and Computer Science

University of Denver

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

---

by

Wassan Hayale

June 2020

Advisor: Mohammad H. Mahoor, Ph.D.

Author: Wassan Hayale
Title: Deep Siamese Neural Networks for Facial Expression Recognition in the Wild
Advisor: Mohammad H. Mahoor, Ph.D
Degree Date: June 2020

Abstract

The variation of facial images in the wild conditions due to head pose, face illumination, and occlusion can significantly affect the Facial Expression Recognition (FER) performance. Moreover, between subject variation introduced by age, gender, ethnic backgrounds, and identity can also influence the FER performance. This Ph.D. dissertation presents a novel algorithm for end-to-end facial expression recognition, valence and arousal estimation, and visual object matching based on deep Siamese Neural Networks to handle the extreme variation that exists in a facial dataset. In our main Siamese Neural Networks for facial expression recognition, the first network represents the classification framework, where we aim to achieve multi-class classification. The second network represents the verification framework, where we use pairwise similarity labels to map images to a feature space where similar inputs are close to each other, and dissimilar inputs are far from each other. Using Siamese architecture enabling us to obtain powerful discriminative features by taking full advantage of the training batches via our pairing strategy, and by dynamically transferring the learning from a local-adaptive verification space into a classification embedding space. These steps enable the algorithm to learn the state of the art features by optimizing the joint identification-verification embedding space. The verification model reduces the intra-class variation by minimizing the distance between the extracted features from the same identity using different strategies. In contrast, the identification model increases the inter-class variation by maximizing the distance between the features extracted from different classes. When a network is tuned carefully, we can rely on the powerful dis-

criminative features to generalize the power of the network to unseen images. Further, we applied our proposed deep Siamese networks on two different challenging tasks in computer vision, valence and arousal estimation and visual object matching. The empirical results of the valence and arousal Siamese model demonstrate that transferring the learning from the classification space to the regression space enhances the regression task since each expression occupies a representation within a specified range of valence and arousal affect. On the other hand, Siamese model of visual object matching gives a better model performance since the classification framework helps to increase the inter-class variation in the verification framework. We evaluated the algorithm using state-of-the-art and challenging datasets such as AffectNet Mollahosseini et al. (2017), FERA2013 Goodfellow et al. (2013), categorical EmotioNet Du et al. (2014), and Cifar-100 Krizhevsky et al. (2009). To the best of our knowledge, this technique is the first to create a powerful recognition system by taking advantage of the features learned from different objective frameworks. We achieved comparable results with other deep learning models.

Acknowledgements

In the beginning, I would like to thank my Lord Allah, who supported me during this long journey towards achieving my highest goal to complete a Ph.D. in Electrical and Computer Engineering.

I would like to express my deepest gratitude and appreciation to my advisor Dr. Mohammad Mahoor for his support, advice, feedback, and guidance. His instructions represented the right direction for all of the problems encountered in the project.

I want to express my sincerest appreciation to my oral defense committee, Dr. Nader Hashemi, Dr. Kimon Valavanis, and Dr. Haluk Ogmen, for their time, consideration, and assistance in improving this project.

I would like to thank all of my colleagues in the Computer Vision lab especially Dr. Pooran Singh Negi who helped and supported me sincerely.

I am also grateful to the University of Denver and all those in charge of it, especially the department of Electrical and Computer Engineering and Anderson Academic Commons buildings for all their facilities.

My thanks to the HCED committee for granting me this scholarship and enabled me to fulfill my dream. Therefore, I will invest this experience in benefiting my country in my area of research.

Finally, I want to dedicate this dissertation to my Parents and my Fiance, who have been the best supporters during the period of study and alienation. I am always getting my strength from their prayers for me.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Automated facial expression recognition (FER) has shown to have positive impacts and influences on our society as it has been widely used in a range of applications. For example, it has been utilized in building the next generation of human-machine interaction (HMI) systems such as driver fatigue surveillance, affect-aware social robotics, and robot-based behavioral therapy. Although extensive research studies have been conducted towards improving FER systems, the variation of facial images in the wild conditions due to head pose, face illumination, and occlusion can significantly affect the FER performance. Moreover, between subject variation introduced by age, gender, ethnic backgrounds, and identity can influence the FER performance Valstar et al. (2012). Hence, more research needs to be conducted to address these challenges.

In the last few years, deep learning has outperformed traditional machine learning methods for visual object recognition. In most of the CNN-based works, softmax function is used along with the cross-entropy loss to train deep models. However, samples within the same class are often dispersed due to the high intra-class variation introduced by the above-mentioned factors. In other words, the intra-class variation can overwhelm the differences between classes and make designing FER systems more challenging.

Several researchers have tried to enhance the discriminative power of softmax. For example, authors in Liu et al. (2017) proposed the angular softmax (A-softmax) to learn angular discriminative features that have smaller maximal intra-class distances than minimal inter-class distances. Other researchers utilized well-designed metrics Cao et al. (2013); Chechik et al. (2009, 2010); Cui et al. (2013); Davis et al. (2007); Globerson and Roweis (2006); Goldberger et al. (2005); Guillaumin et al. (2009); Lu et al. (2013); Nguyen and Bai (2010); Qamar and Gaussier (2009); Qamar et al. (2008); Shalev-Shwartz et al. (2004); Weinberger et al. (2006); Xing et al. (2003); Zheng et al. (2011) instead of softmax to tackle the intra-class variation by learning distance/similarity metrics such that the features of the images belonging to the same label become close to each other, while images of different identities become far from each other. However, one fundamental limitation of these approaches is that they are often linear and shallow, and rely on hand-crafted image descriptors, while the inter- and intra-class variation are non-linear and observed only in a high-dimensional space Sun et al. (2014). Alternatively, with the outstanding performance of deep learning, non-linear mappings can be achieved with CNNs to automatically learn discriminative features directly from the samples followed by a simple distance metric such as Euclidean. In this dissertation, we adopt the deep metric learning as a non-linear transformation to embed images into the feature space.

There are also several works on jointly training a traditional metric loss using softmax. For example, authors in Sun et al. (2014) presented the idea of a joint identification-verification representation but for a verification task rather than an identification task. In their architecture, the features are learned by using both signals simultaneously. Their work differs from our approach in a way that both losses are weighted by a hyperparameter. Another work McLaughlin et al. (2016) explicitly used a joint identification-verification system, but it weighted both costs equally. Therefore, the network was trained to satisfy both objectives. They found that jointly optimizing both costs is crucial for convergence.

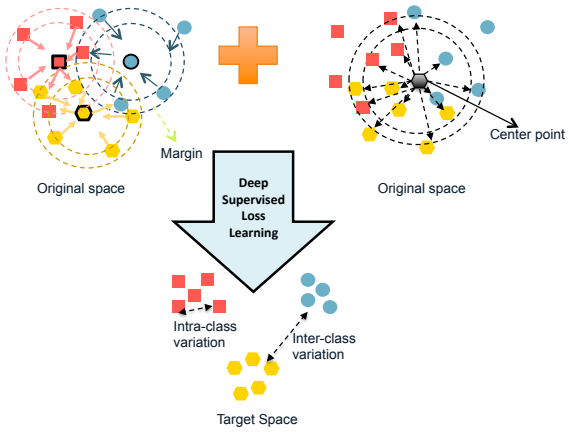


Figure 1.1: An intuitive illustration for constructing ideal features representation of the face recognition system. Reducing the distance between the examples and increasing it between the examples and center point (zero vector, local mean, or global mean) leads to low intra-class variation and high inter-class variation.

We argue that we can do better by adopting deep convolutional Siamese networks that are controlled by two supervisory loss functions. The verification loss is combined into the identification loss such that its early contribution to the total loss function can assist in decreasing the intra-class variation. After training the system for several epochs, we decay the verification loss until it vanishes, keeping the identification loss [1] targeting the inter-class variation.

Therefore, to create a reliable and robust supervised metric learning method, the training procedure should handle the extreme variation that exists in the facial dataset. Additionally, it should be able to integrate a multi-objective loss function to manage the intra/inter-class variation without increasing the training computational complexity.

In summary, this dissertation proposes a novel representation based on supervised loss function by using deep Siamese Neural Networks, aiming to decrease the intra-class variation and increase the inter-class variation to enhance the FER performance, as shown in Figure 1.1.

[1] To be consistent with the literature in face recognition; we refer to classification as identification throughout this dissertation.

Our major contributions are summarized as follows:

1. A novel deep Siamese Neural Networks that contain an identification framework and a verification framework.

2. Integrating a multiple-objective loss function in the verification framework to enhance the discriminative power of the verification embedding space.

3. Ability to transfer the learning from the verification space to the identification space to enhance the discriminative power of the identification embedding space.

4. Introducing a novel mining strategy to handle the negative pairs constraints without additional computational resources and memory requirements.

5. Adapting the local structure for the embedding space by a novel pairing strategy.

The remainder of this dissertation is organized as follows: Chapter 2 reviews the related literature on metric learning and Siamese Neural Networks. Chapter 3 describes our proposed Siamese Neural Networks architecture, which includes the identification framework and verification framework. Our data mining strategy and a multi-task metric used in the verification framework are also described in this chapter. Chapter 4 introduces an end-to-end algorithm for the valence and arousal estimation using Siamese Neural Networks. Chapter 5 describes one of Siamese Neural Netowrk applications of image matching with Cifar-100 dataset. Finally, Chapter 6 concludes the dissertation with some discussions and suggestions for future work.

Chapter 2

Related Work

Image-based facial expression recognition systems have been actively studied in the past few years. Since this dissertation is mainly concerned with developing a metric-based cost function and Siamese Neural Networks, we briefly review the related works in these two areas.

2.1 Siamese Neural Networks

Siamese Neural Networks (SNNs) comprise of two identical networks that share the weights and are joined by an energy function at the top. One of the efficient energy functions used alongside with SNNs is a contrastive energy function, which contains dual terms to decrease the energy of similar pairs and increase the energy of dissimilar pairs.

Siamese Neural Networks are first introduced in 1994 by Bromley Bromley et al. (1994), where the authors described an algorithm for a signature verification that is written on a pen-input tablet. Since then, SNNs have drawn great attention in many computer vision applications. One of the most fundamental problems in computer vision that uses SNNs is visual tracking problems Fan and Ling (2019); Li et al. (2019); Zhang and Peng

(2019). It aims to estimate the position of an arbitrary target in a video sequence, given only its location in the initial frame. They formulated object tracking as a matching problem where a Siamese model aims to learn a similarity function from a large set of data.

Another example is a person re-identification problem Zheng et al. (2019), where a Siamese architecture is trained to predict the similarity or the distance between two input images. Moreover, one can find the Siamese Neural Networks trace in video face recognition Yang et al. (2017), face verification Taigman et al. (2014), localization Tompson et al. (2015), image descriptors Kumar et al. (2016), zero-shot recognition Kiran Yelamarthi et al. (2018), one-shot recognition Koch et al. (2015), few-shot recognition Sung et al. (2018), signature verification Wu et al., sentence similarity Mueller and Thyagarajan (2016), text similarity Neculoiu et al. (2016), detection Klomp et al. (2019), image segmentation Lu et al. (2019), and image retrieval Qi et al. (2016).

In this dissertation, we create Siamese networks architecture comprising of two frameworks: a recognition framework and a verification framework. Specifically, we transfer the learning from the verification framework into the recognition framework. Each framework or branch has its own loss function. The identification framework has a cross-entropy loss function to classify facial emotion into several classes. On the other hand, the batch of inputs in the verification framework is resampled as a similar/dissimilar pairs in order to build a similarity loss function that takes those pairs of input and measure the distance/similarity between them.

2.2 Metric Learning

Learning an effective metric for the estimation of discriminative features is the core of a multi-class classification task as it draws the objective of the system. According to different linear metrics used on feature vectors, one can divide linear Metric Learning into

two main categories: **distance metric learning** and **similarity metric learning**. Despite their conceptual differences, these two types are formed based on a similar goal. Specifically, to learn a metric such that more discriminative information can be exploited via mapping features to an embedding space where intra-class samples are close to each other, and inter-class samples are pushed away from each other.

Distance metric learning, as the name suggests, uses the Euclidean distance to measure the pairwise relationship between two feature vectors. To name a few, authors in Cui et al. (2013); Davis et al. (2007); Globerson and Roweis (2006); Goldberger et al. (2005); Guillaumin et al. (2009); Lu et al. (2013); Shalev-Shwartz et al. (2004); Weinberger et al. (2006); Xing et al. (2003); Zheng et al. (2011) proposed algorithms to minimize the Euclidean distance between the similar pair and to separate the dissimilar pair with a large distance. On the other hand, authors in Cao et al. (2013); Chechik et al. (2009); Li et al. (2013); Nguyen and Bai (2010); Qamar and Gaussier (2009); Qamar et al. (2008) used the similarity measure to map the images into a discriminative embedding space. For example, Qamar et al. (2008) formulated the similarity metric learning using Cosine similarity. They optimized the similarity metric using an online training approach, and they showed that the Cosine similarity is preferred over the Euclidean distance on several data collection sets. Along a different line, Cao et al. (2013) developed Similarity Metric Learning by incorporating both a Mahalanobis distance metric and a bilinear similarity metric, simultaneously. The formulation was proven to be a convex optimization problem that guarantees the existence of its global solution.

The above linear metric learning approaches may suffer from non-linear correlations of samples. Alternatively, as the discriminative power of kernel trick methods is limited, deep metric learning approaches Bhattarai et al. (2016); Cui et al. (2016); Duan et al. (2018); Harwood et al. (2017); Hu et al. (2014, 2015); Huang et al. (2016); Iscen et al. (2018); Kumar et al. (2016); Liu et al. (2017); Lu et al. (2013, 2015); Movshovitz-Attias et al.

(2017); Oh Song et al. (2016, 2017); Paisitkriangkrai et al. (2015); Schroff et al. (2015); Sohn (2016); Taigman et al. (2014); Ustinova and Lempitsky (2016); Wang et al. (2014a, 2017); Wen et al. (2016b) have been proposed to learn a non-linear mapping directly from the samples. Researchers have adopted five major approaches in this area to achieve a better performance.

In the first approach, facial images are used along with their identity labels to learn discriminative identification features in a classification framework. Most of these algorithms use a Deep Convolutional Neural Network (DCNN) architecture along with softmax to learn discriminative identification features. For Example, VGG-Face model Parkhi et al. (2015) used various Convolutional Neural Network (CNN) architectures for face identification and verification tasks.

In the second approach, researchers tried to enhance the discriminative power of softmax function. For example, the authors in Liu et al. (2017) proposed the angular softmax (A-softmax) to learn angular discriminative features that have smaller maximal intra-class distance than the minimal inter-class distance.

In the third approach Hu et al. (2014); Lu et al. (2013, 2015), pairs of images (genuine or impostor) along with the same or not-same label are used to learn a feature embedding where genuine pairs are closer, and impostor pairs are far apart. In this approach, the total loss (constructive loss) is composed of two partial loss functions, one for the genuine pair and the other for the impostor pair. It is designed in such a way that the minimization of total loss reduces the distance for genuine pairs and increases the distance of impostor pairs. Along similar line, the authors in Duan et al. (2018); Iscen et al. (2018); Movshovitz-Attias et al. (2017); Schroff et al. (2015); Ustinova and Lempitsky (2016); Wang et al. (2014a) introduced a triplet loss to learn the metric using triplet face samples while others such as Huang et al. (2016) moved towards quadratic loss to learn the metric using four face samples.

The fourth approach is similar to the third approach except that instead of computing the similarity score between pair, triple, or quadratic samples, it learns the embedding space by comparing more comprehensive pairs that reflect the local or global structure of the embedding space. For example, the authors in Oh Song et al. (2016) took the full advantage of training pair examples through introducing a matrix of a pairwise distance within a batch instead of a vector of a pairwise distance. The authors in Song et al. (2017) proposed learning an end-to-end framework that is aware of the global structure of the embedding space and designed to optimize the clustering quality.

In the fifth approach, researchers moved towards joint training of multiple objectives loss in which they developed an effective feature representation by using multiple signals as a supervision simultaneously. For example, the authors in Wen et al. (2016b) enhanced the discriminative power of the learned features by proposing another supervising signal, named a center loss, to supervise the deep model along with softmax function. The center loss aims to learn a center for the deep features of each class and minimize the distances between the class features and corresponding class centers. Authors in McLaughlin et al. (2016); Sun et al. (2014); Wen et al. (2016a,b) explored using deep learning along with the face identification and verification signals as supervision.

In this dissertation, we adopt the fifth approach of joint identification-verification representation, but we enhance it with the idea of transfer the learning from the verification system to the identification system. Therefore, instead of weighting both objectives equally as in McLaughlin et al. (2016) or using a hyperparameter to weight them as in Sun et al. (2014), we propose a weighting scheme by which the verification signal contributes at the early training time. Later this signal vanishes while the identification signal is involved gradually in the system. In our previous work Hayale et al., we presented a method for embedding two signals, the verification and identification, into a facial expression recognition system using a deep Siamese Neural Networks. We obtained promising results by

dynamically modulating the verification signal over the identification signal. We further investigate that idea by integrating a metric that reflects the local structure of the embedding space and by using a novel pairing strategy to handle negative pairs constraints without extra computational burdens and memory requirements.

Chapter 3

Facial Expression Recognition using Siamese Neural Networks

Figure 3.1 illustrates a diagram of our proposed Siamese networks architecture. It has two identical networks sharing the same parameters. The two identical networks perform a non-linear mapping from the input domain to a high-level embedding space, which is responsible for creating more discriminative features. For each network, we use the state-of-the-art DenseNet architecture presented in Huang et al. (2017), although other deep networks can be used.

The first network represents the classification framework, where we aim to achieve multi-class classification using a cross-entropy loss function along with softmax function. While the second network represents the verification framework, where we use the same or not same labels to map images to a feature space where similar inputs are close to each other, and dissimilar inputs are far from each other. Our first goal is to achieve a metric learning scheme through the verification framework that is aware of the local structure of the embedding space through a novel pairing strategy. The second goal is to achieve a multi-task metric learning approach in the verification framework in which we will lever-

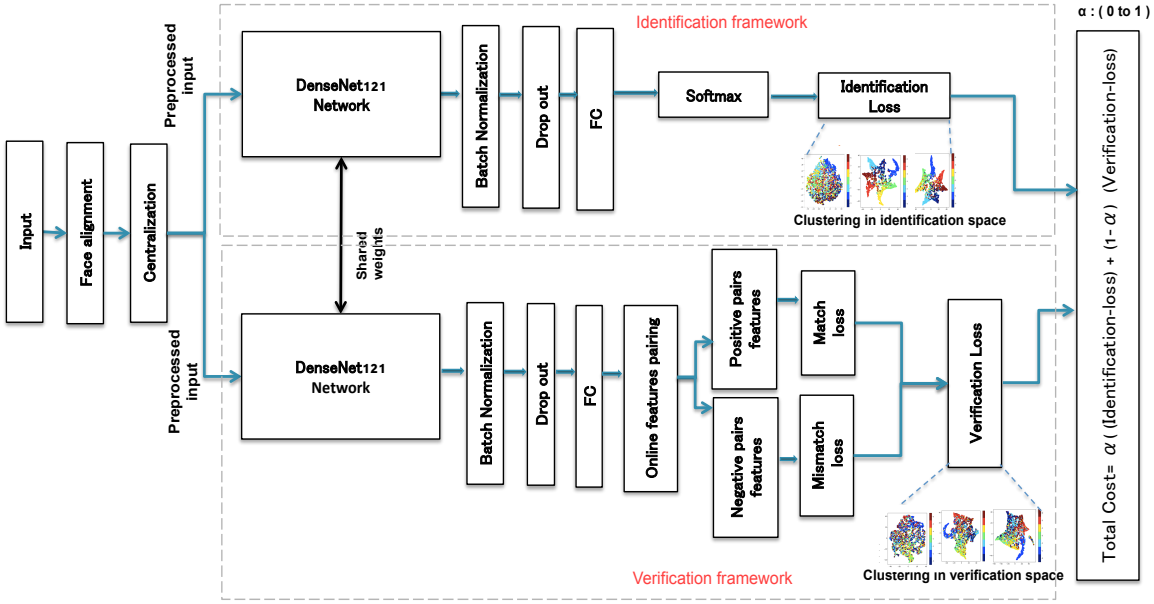


Figure 3.1: Main diagram of our proposed architecture.

age the performance of a single task by combining multiple objective losses. The final features can be more discriminative by using both face identification and verification signals as supervision signals. In order to achieve that, our third goal is to enable the transfer learning from the high dimensional features within the verification space to the identification space to enhance the identification task objective. Finally, our fourth goal is to exploit multiple visual features by building an ensemble of various models, in which each model has a different verification framework. The final model performance is calculated from the average sum of these model predictions.

We train our network using AffectNet Mollahosseini et al. (2017), FER2013 FER, and categorical EmotioNet Du et al. (2014) datasets and derive the joint identification-verification metric using the DCNN features of the training data. Then, given a facial expression image, we classify the image into different essential expressions based on their DCNN features and the learned metric. We will provide further details of each component of our architecture in the following subsections.

3.1 Identification Framework

The first branch in our network is the identification framework which is achieved by feeding the features to softmax layer. This layer classifies each image into one of seven different expressions by giving the probability distribution over the seven classes. Given a feature vector f with its associated emotion label y, we can derive the probability distribution as follows:

$$P_i = P(y = i|f) = \frac{\exp^{W_i f}}{\sum_k \exp^{W_k f}} \tag{3.1}$$

where W_i refers to the i[th] column of softmax weight matrix. The network is then trained to minimize the cross-entropy loss (identification loss) as defined below:

$$IdentLoss = -\sum_{i=1}^{K} y \log P_i \tag{3.2}$$

where $y = 0$ stands for all classes except for the target class, and K represents the total number of examples in the dataset. Since the AffectNet and FER2013 datasets are heavily imbalanced, we utilize the weighted-loss approach presented in Mollahosseini et al. (2017), while we use the regular cross-entropy loss for EmotiNet-7 and EmotiNet-22 datasets. The weighted-loss approach weights the loss function for each class according to its relative proportion in the dataset. In other words, the loss function sets different penalization weights for under-represented classes and well-represented-classes. We can define the weighted loss for an identification signal as follows:

$$Ident_{weighted} = -\sum_{i=1}^{K} y H_i \log P_i \tag{3.3}$$

where H_i represents penalization weight for class i.

3.2 Verification Framework

Traditionally, the usual loss functions used for verification learning are contrastive loss Chopra et al. (2005); Hadsell et al. (2006) or triplet loss Dong and Shen (2018); Schroff et al. (2015). Therefore, the training procedure in those approaches relies on creating positive pairs (similar samples having similar class labels) and negative pairs (dissimilar samples having different class labels). We briefly introduce our pairing strategy in Section 3.2.1 and the proposed metric for this framework in Section 3.2.2.

3.2.1 Data Mining and Local Structured Pairing

In the verification framework, we train a deep neural network to learn a set of hierarchical non-linear transformations to project facial images into a feature space, under which the distance of each positive pair is reduced, and the distance of each negative pair is enlarged. Therefore, the proposed loss function accepts a pair of positive and negative features to optimize the similarity or the distance between them.

Consider Pos, and Neg are two sets of all possible positive and negative pairs that can be generated from the training images that have M categories. Assuming each class has N number of images, then the total number of pairs are:

$$Pos = \sum_{i=1}^{M} N_i * (N_i - 1) \tag{3.4}$$

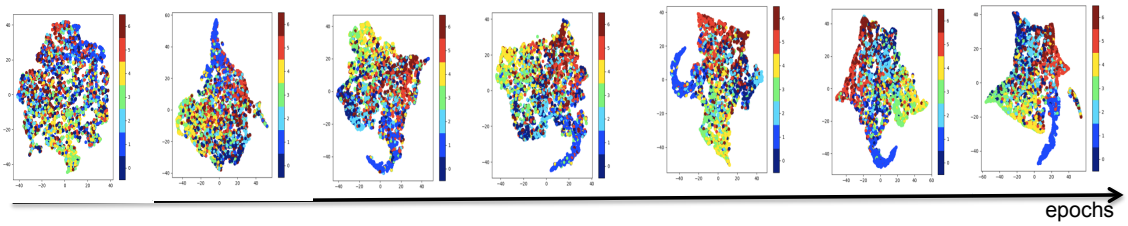
$$Neg = \sum_{i=1}^{M} \sum_{j=1}^{M} N_i * N_j \qquad i \neq j \tag{3.5}$$

With a large volume of data pairs, this results in a slow training convergence. Previous works Harwood et al. (2017); Iscen et al. (2018); Oh Song et al. (2016); Schroff et al. (2015); Sohn (2016); Song et al. (2017) have shown that mining strategy plays an essential

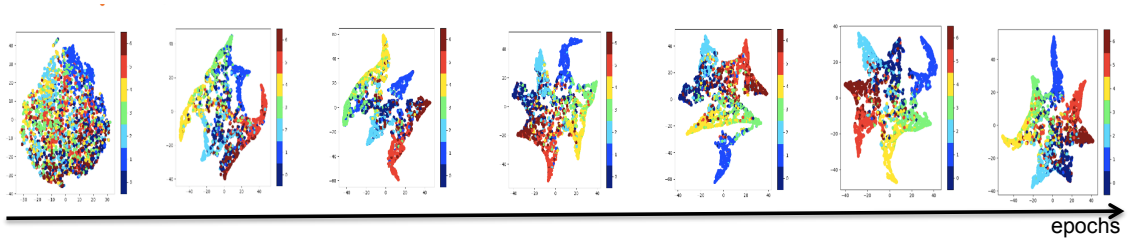
role in Siamese networks training as it relies on only hard negative examples and hard positive examples to produce gradient with a sufficiently large magnitude. On the other hand, it can be challenging to determine the "best" representative pairs that can contribute most to the network convergence. Several approaches tried to tackle these issues through learning by stochastic gradient descent (SGD) or (online learning) in which they considered the pairs in each iteration of batches Chechik et al. (2009). Other by learning with carefully designed pairs, in which all the negative and positive pairs were prepared in advance, which is very time consuming and difficult to accomplish with large-scale dataset Hadsell et al. (2006). While other Harwood et al. (2017); Iscen et al. (2018) tended to use a mining strategy for negative samples in which they picked only the hard negative examples that produce gradient with large magnitudes.

Before explaining our pairing strategy, we need to have good insight into the feature space corresponding to these two signals. We use a t-distributed stochastic neighbor embedding (t-SNE)[1] to visualize the mentioned feature space as shown in Figure 3.2. Figure 3.2a shows the verification signal with an objective of making the samples sharing the same class labels close to each other, and pushing the samples that have different class labels away from each other. It shows that the verification signal has succeeded in reducing the intra-class variation, but it did not enlarge the inter-class variation well. In contrast, using only an identification signal as shown in Figure 3.2b can enlarge the inter-class variation significantly Sun et al. (2014). Since the inter-class variation can be optimized via negative pairs, using an identification signal compensates for the use of negative pairs in the verification framework. However, instead of neglecting all the negative pairs in the verification framework, the mining strategy is applied to the negative pairs in which we generate negative pairs between only the classes that are close to each other. To explore

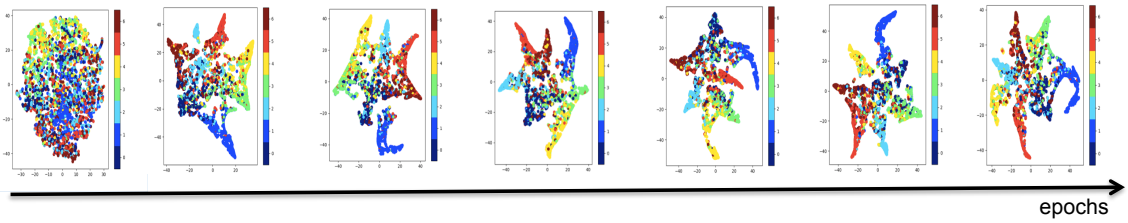
[1]t-SNE is a technique used for a non-linear dimensionality reduction for embedding high-dimensional data into two or three dimensions space, that can then be visualized in a scatter plot.



(a) Verification Signal



(b) Identification Signal



(c) Both signals

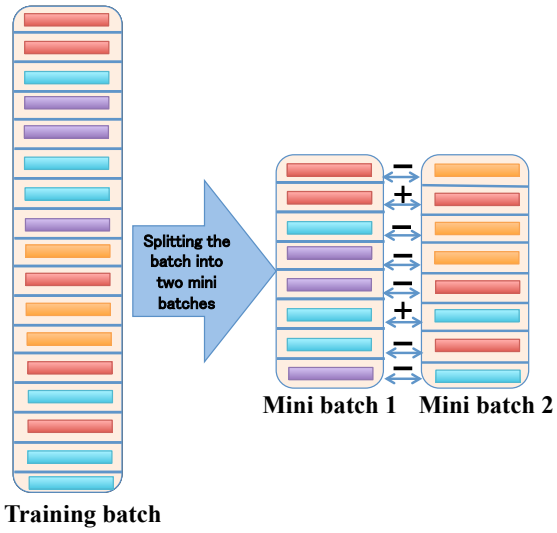
Figure 3.2: Visualization of 2-dimensional features for seven classes of AffecNet dataset across several epochs. The indices 0 through 6 represent Neutral, Happy, Sadness, Surprise, Fear, Disgust, and Angry classes, respectively. (a) Illustrates the feature space when using only the verification signal. It has small intra-class variation; on the other hand, the inter-class variation is also small, which causes different classes to be close to each other. (b) Illustrates the feature space using only the identification signal. There is a large inter-class variation; on the other hand, the intra-class variation is not too small, which causes the identities that belong to the same class to be far apart from each other. (c) Illustrates the feature space using both signals where both inter- and intra-class variation are enhanced eventually.

the classes that are close to each other, we train a single DenseNet network for multi-class classification. From Figures 3.2b and 3.7, we can observe that several classes are closer to each other than other classes (e.g., Neutral with Sadness, Surprise with Fear, and Disgust with Angry).

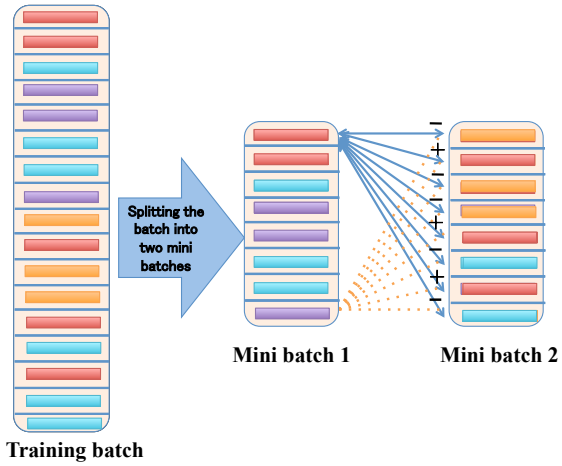
3.2.1.1 Full pairing strategy

In this dissertation, instead of the offline pairing approach in which the pairs are built and fed as a batch of input pairs into two identical networks, we follow the online pairing approach. We feed a batch of single examples to our verification network then we generate the pairs from the features directly at the embedding space. Specifically, after a non-linear mapping to the embedding space, each batch of features will be split into two mini-batches. Afterward, we generate positive features and negative features from these two mini-batches by taking each feature from one mini-batch with the corresponding feature from the other mini-batch, as shown in Figure 3.3a. We name this approach as an **Element-wise** approach.

Further, we extend this approach to reflect the local structure of the embedding space by lifting the pairs vector and taking each example from one mini-batch with all examples from the other mini-batch, as shown in Figure 3.3b. We name this approach as **Pair-wise** approach. By leveraging the ideal properties of generated positive and negative pairs, the training efficiency can be improved, the learned metric becomes robust to intra/inter-class variation, and the computational cost of offline pairing can be dramatically reduced through the online pairing. Since both approaches utilize prediction-to-prediction pairing in which both predictions can be obtained from the current training batch, we designate both approaches as a **Full Pairing** strategy.



(a) Element-wise mini-batches pairing



(b) Pair-wise mini-batches pairing

Figure 3.3: Full pairing strategies.

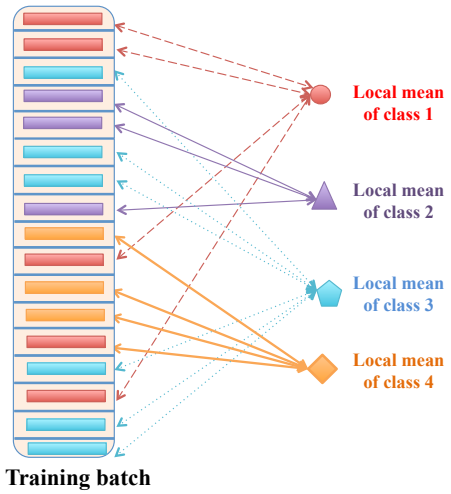
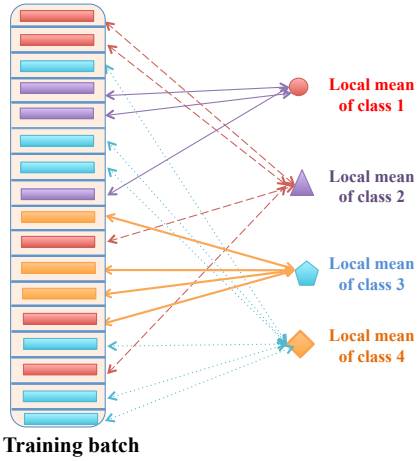


Figure 3.4: Partial positive pairing strategy.

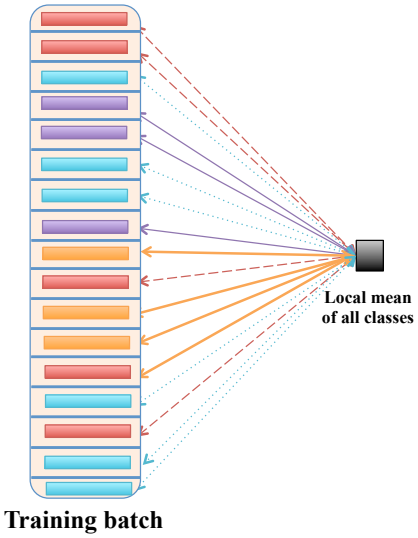
3.2.1.2 Partial pairing strategy

We consider another two pairing strategies where the local mean of each class and the mean of all classes are involved along with all examples in the batch. We name this strategy as **Partial Pairing** strategy as it does not include two predictions in the generated pairs, and it does not require splitting the batch into two mini-batches. For the first partial strategy, we consider generating pairs by pairing each example with its local class mean for positive pairs, as shown in Figure 3.4, and with the local class means of the other classes for negative pairs as shown in Figure 3.5a. However, instead of considering all the possible negative pairs between the classes, we consider only the examples for classes close to each other since the far classes will not produce a potential gradient. We refer to this pairing strategy as **Partial-1**.

The second partial strategy involves the pairing between all the examples and the local mean of all the examples for negative pairs, as shown in Figure 3.5b, along with the same **Partial-1** positive pairing strategy. We refer to this pairing strategy as **Partial-2**. For this pairing strategy, we also investigate a global mean and a zero-vector instead of the local mean. For better comparison in Experiment and Results section in Section 3.4, we name



(a) Partial-1: prediction to class mean negative pairing.



(b) Partial-2: prediction to local mean negative pairing.

Figure 3.5: Partial negative pairing strategies.

the above approaches as **Local-mean Partial-2**, **Global-mean Partial-2**, and **Regularized Partial-2**. In summary, our mining strategy includes online pairing that reflects the local structure of the embedding space besides mining the negative pairs to be only between the classes that need to be separated by a specified margin.

3.2.2 Multi-task Metric Learning

3.2.2.1 Full-based pairing loss function

We use Euclidean distance and Cosine similarity functions or the combination of both to design our metric. Since the proposed architecture involves many individual processing steps, for notational simplicity, we refer to the complete network as a function, $f = F(x)$, that takes an image x as an input and produces a vector f as an output. In such case, the Euclidean distance D and Cosine similarity S between two vectors $f_1,\ f_2 \in \mathbb{R}^d$ can be written as:

$$D^2(f_1, f_2) = \|(f_1 - f_2)\|^2 \tag{3.6}$$

$$S(f_1, f_2) = \frac{{f_1}^T f_2}{\|f_1\|\|f_2\|} \tag{3.7}$$

We propose a multi-task metric learning method to simultaneously learn an Euclidean distance and a bilinear similarity between all the examples along with the Euclidean distance between the examples and the local mean of the data. Therefore, to learn an improved metric function for the verification task containing both, we adopt an energy-based local-adaptive similarity metric learning algorithm, which is defined as:

$$\begin{aligned}\text{Loc-SML}(f_1, f_2, f_{avg}) = S(f_1, f_2) - D(f_1, f_2) \\ + 1/D(f_1, f_{avg}) + 1/D(f_2), f_{avg})\end{aligned} \tag{3.8}$$

where f_{Avg} represents the local mean of the batch. We call this method a **Loc-SML**, short for Similarity Metric Learning enhanced with local structure. Specifically, it learns jointly, (i) the Euclidean distance as a projection into a low dimensional Euclidean space, (ii) Cosine similarity as a projection into a low dimensional Similarity space, (iii) Euclidean distance between all the examples and the local mean of the examples. By jointly learning multiple objectives, the information shared between the related tasks can lead to improved performance.

This Loc-SML loss function is composed of two items, a match loss which acts as a pulling force and a mismatch loss which acts as a pushing force.

$$J_{match} = \sum_{f_i f_j \in Pos} max(0, 1 - \text{Loc-SML}) \tag{3.9}$$

$$J_{mismatch} = \sum_{f_i f_j \in Neg} max(0, 1 + \text{Loc-SML}) \tag{3.10}$$

Unlike other methods that usually take pairs in the Pos and Neg sets for the whole training set, Loc-SML contains only similar and dissimilar pairs created from the training batch by following our novel pairing strategies. More details about our pairing strategies are mentioned in Section 3.2.1.

Let $label = 1$ (respectively $label = 0$) denotes the pairs being similar (respectively dissimilar), then the total Loc-SML cost is defined as :

$$J_{total} = label * J_{match} + (1 - label) * J_{mismatch} \tag{3.11}$$

which can be explained as two constraints:

- When the label = 1, we have $1 -$ Loc-SML ≤ 0 which means the Loc-SML of a similar pair should be larger than 1.

- When the label = 0, we have $1 + \text{Loc-SML} \leq 0$, which means the Loc-SML of a dissimilar pair should be smaller than than -1.

Higher energy indicates high S and low D, which suggests f_1 and f_2 form a genuine pair. On the contrary, low energy indicates low S and high D, which suggests f_1 and f_2 form an imposter pair. Clearly, the term $1/D$ is not used for the J_{match} as it reflects only the pushing force in our loss function.

For the first two terms in Equation 3.8 we use the **Element-wise** full pairing strategy for the positive and negative pairs (See Figure 3.3a). While for the third and fourth terms in Equation 3.8, we pair all the examples with the local mean of the data to maximize the distance between them following the pairing strategy shown in Figure 3.5b. We refer to this model as **Full-LocalMean** model.

For the third and fourth terms Equation 3.8, we also investigated using global mean (the mean of the whole training set) and a zero vector instead of the local mean. We refer to these models as **Full-globalMean** and **Full-regularized**, respectively.

3.2.2.2 Partial-based pairing loss function

We further apply our **Partial pairing strategy** mentioned in Section 3.2.1.2. In the **Partial-1** model, given a batch of inputs to the verification framework, each batch is processed to generate mini-batches $F_1, F_2, \ldots, F_k$, for $k \in \{1, \ldots, K\}$, which represents the mini-batches of features belonging to each class from K total number of classes. Then an average feature f_{avg} is calculated for each class. Hence, we can write the Siamese networks training objective for the positive pair's loss as follows:

$$Partial_{match} = \sum_{k=1}^{K} \frac{1}{2} \| F_k - f_{avg}^{(k)} \|^2 \tag{3.12}$$

where K is the total number of classes in the dataset, $\|F_k - f_{avg}^{(k)}\|$ is the Euclidean distance between the mini-batch features of one class and average feature for that class. The objective encourages the features to be close to their class mean. In contrast, we define the objective for the negative pairs loss function to enforce the features belonging to close classes to be separated by a margin m as follows:

$$Partial1_{mismatch} = \sum_{k=1}^{K} \frac{1}{2}[max(m - \|F_k - f_{Avg}\|, 0)]^2 \tag{3.13}$$

where K it the total number of classes in the dataset, $\|F_k - f_{Avg}\|$ is the Euclidean distance between the mini-batch features of one class and average feature for the class that is close to a class F_k. The margin m is set to 2 in all our experiments.

In **Partial-2** model, we define the objective for the negative pairs loss function to maximize the distance between the features exist in the batch and the local mean of the batch. This can be written as:

$$Partial2_{mismatch} = \sum_{i=1}^{N} \frac{1}{2}[max(m - \|f_i - f_{Avg}\|, 0)]^2 \tag{3.14}$$

where N is the total number of features in the batch, $\|f_i - f_{Avg}\|$ represents the Euclidean distance between all the features in the batch and the local mean of the batch.

We can drive our final Partial loss function for verification framework as

$$Partial_{total} = label * Partial_{match} + (1 - label) * Partial_{mismatch} \tag{3.15}$$

where $Partial_{mismatch}$ involves either $Partial1_{mismatch}$ or $Partial2_{mismatch}$ loss function. $label = 1$ (respectively $label = 0$) denotes the pairs being similar (respectively dissimilar).

3.3 Joint Identification-Verification

The learned features in the identification space are revealed to have a large inter-class variation as shown in Figure 3.2b. This is attributed to the identification supervisory signal that tends to pull apart the features of different identities into seven different classes. On the other hand, in order to generate an effective identification system, we need to enhance the system with the verification signal, which has shown that it emphasizes the intra-class variation. We noticed that the early activation of the verification signal improve the clustering of dataset classes. Consequently, after several epochs, the verification signal vanishes through a controlled hyperparameter, while the identification signal is activated gradually. Thus, we can define the overall training loss function, which jointly optimizes the verification cost and the identification cost as follows:

$$Total_{loss} = (1 - \alpha).Verif_{loss} + \alpha.Ident_{weighted} \qquad \alpha : 0 \rightarrow 1 \tag{3.16}$$

$$\alpha = \frac{1}{\exp \frac{epoch_{num} - shift}{10}} \tag{3.17}$$

where $Verif_{loss}$ includes one of the verification models discussed in Section 3.2.2. The $shift$ factor represents the epoch number at which the $Verif_{loss}$ and $Ident_{loss}$ have equal contributions in the $Total_{loss}$. After that, the $Verif_{loss}$ begins to vanish while the $Ident_{loss}$ involves gradually into the network training. Thus we found that satisfying both the verification and identification losses by jointly training is crucial for convergence.

Lastly, besides our model in Equation 3.8, we investigate different models that contain either the Euclidean distance or the Cosine similarity. Following that, we employ an ensemble method to exploit the relations between the tasks and potentially improve the performance. For Euclidean distance, we follow the approach in McLaughlin et al. (2016),

which can be written as:

$$Loss_{Euc}(f_i, f_j) = \begin{cases} D(f_i, f_j)^2 & i = j \\ \frac{1}{2} max(0, m - D(f_i, f_j))^2 & i \neq j \end{cases} \tag{3.18}$$

In the same manner, the first part of this equation aims to minimize the distance between the similar pairs in the target space, while the second part seeks to separate the dissimilar pairs with a margin m. We refer to this model as **Euclidean** model.

While for the Cosine similarity we follow the approach mentioned in Qamar and Gaussier (2009) in which it can be written as :

$$Loss_{gCosLA}(f_i, f_j) = \begin{cases} S(f_i, f_j) \geq b + \gamma & i = j \\ S(f_i, f_j) \leq b - \gamma & i \neq j \end{cases} \tag{3.19}$$

Ultimately, the underlying assumption of gCosLA loss is that the similarity between the similar pairs should always be larger than the similarity between the dissimilar pairs. More precisely, to enhance that assumption, another threshold b has been introduced along with the margin γ as a specified margin between the dissimilar pairs in the target space. We refer to this model as **gCosLA** model. Consequently, a full comparison between all the models was mentioned in Section 3.4.

Later on, we exploit multiple visual features by building an ensemble of previous models, in which each model has a different verification framework. The best performance due to different pairing strategies for each model will be combined with the best performance for our original Loc-SML model. The model performance is calculated from the average sum of these model predictions.

3.4 Experiments and Results

This section presents the evaluation results of the proposed architecture for the facial expression identification task using the AffectNet, FER2013, and compound EmotiNet datasets. We introduce the datasets, implementation details for our network, and the experiments in the following subsections.

3.4.1 Datasets

The **AffectNet** dataset Mollahosseini et al. (2017) is the largest database of the facial expression that provides a dimensional and categorical representations of emotion. The database contains about 1M facial images, which is generated by querying several search engines and manually annotated for the basic facial expressions. Since the test set is not released, the evaluation protocol is determined according to the provided validation set.

The **FER2013** FER database contains a total of 32,295 images of different identities and 3,592 for the test set. Seven classes are defined in this dataset (i.e., six basic emotions plus neutral). The size of the images is set to 48×48 pixels. The dataset is created using the Google image search API, and all the faces are automatically registered so that the faces are centered in the images.

The **compound emotion categories (EmotioNet-22)** Du et al. (2014) dataset is created by combining the basic component categories. Therefore, 22 categorical emotions are defined, including six basic emotions and 15 compound facial expressions of emotions. For example, a happily surprise emotion is constructed using a happy with a surprise component throughout combining the muscle movements observed in happiness and surprise. The 22 facial expressions are collected from 230 human subjects. Most ethnicities and races are represented in the database. A computational model is driven for key facial point automatic detection, which defines the shape and external/internal features of the

face. Then, the automatic categorization of basic and compound emotions is utilized using shape and appearance features. First, we report the results on the six basic emotions plus the neutral using 1,610 images of the 230 identities. Then we calculate the accuracy for the 5,060 images corresponding to the 22 categories of the basic and compound emotions (plus neutral) from 230 identities in the dataset.

3.4.2 Implementation Details

We use DenseNet architecture as our deep CNNs baseline, though other networks can be utilized. The DenseNet consists of five blocks in which each block has several convolution layers, and each layer is connected to every other layer in a feed-forward fashion. Thus, each layer receives the feature-maps of all preceding layers. More details about the baseline architecture are described in Huang et al. (2017).

The AffectNet faces are cropped from images using the bounding box provided in the AffectNet metadata file, while the faces of other datasets are already cropped. The faces of all datasets are resized to 256×256 pixels except AffectNet dataset, where the faces are resized to 128×128 pixels. Additionally, landmark-based face alignment is performed on AffectNet dataset, while we do not perform any type of alignment on other datasets as they are already aligned enough. We perform per-image standardization on all datasets that linearly scales each image to have zero mean and variance equal to one.

Six types of augmentations, such as flip, brightness, contrast, rotation, hue, and saturation, are applied to create more training samples. The network is trained using a batch size of 32 for all dataset except AffectNet dataset, where the batch size is set to 128. We use Adam optimizer Kingma and Ba (2014) as an adaptive learning rate optimization algorithm for training our deep neural network. The baseline learning rate is set to 0.001 and decreased by a factor of 0.1 when the metric stops improving after each ten epochs.

To get the best computational performance, we use TensorFlow as the most popular and efficient machine learning tool for training our network. Keras is also used as a high-level neural networks API (Application Program Interface) that wraps a sequence of complicated underlying TensorFlow operations. Moreover, we run our experiments on two NVIDIA 1080 Ti GPUs (Graphics Processing Unit) as underlying computing devices.

3.4.3 Experiments

We present the results of different experiments on the three datasets to demonstrate the effectiveness of our purposed model, which includes our verification framework and our pairing strategy. We finally compare with state-of-the-art methods for the three datasets.

3.4.3.1 Different identification losses

In the first experiment, we investigate how the weighted-loss affects the learned features of the under-represented classes. First, we train a single identification network without using the weighted-loss. Afterward, in the weighted-loss approach, we set different penalization weights for the under-represented classes and well-represented-classes in AffectNet and Fer2013 datasets. We do not use the weighted-loss on EmotioNet-7 and EmotioNet-22 datasets as they are balanced datasets. Table 3.1 shows that the accuracy increased for AffectNet and FER2013 datasets. It can be noticed from the confusion matrices in Figures 3.6, 3.7, 3.8, and 3.9 that the facial expression recognition validation accuracy for both datasets increased comparing to the imbalanced approach. The performance for the under-represented classes like fear and disgust is better with a weighted-loss approach as this approach penalizes more the misclassified examples from these classes.

Table 3.1: Validation accuracy (%) with weighted/un-weighted loss for identification framework.

Methods	AffectNet	FER2013	EmotioNet-7	EmotioNet-22
Un-weighted loss	56.45 $\pm$0.21	64.48 $\pm$0.04	**97.63$\pm$0.23**	**69.29 $\pm$0.58**
Weighted loss	**61.01 $\pm$0.22**	**67.89 $\pm$0.20**	N/A	N/A

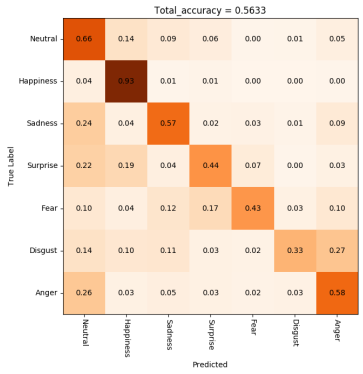


Figure 3.6: AffectNet validation confusion matrix for single identification DenseNet network without weighted-loss approach.

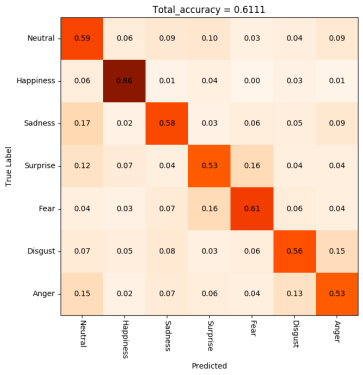


Figure 3.7: AffectNet validation confusion matrix for single identification DenseNet network enhanced with a weighted-loss approach.

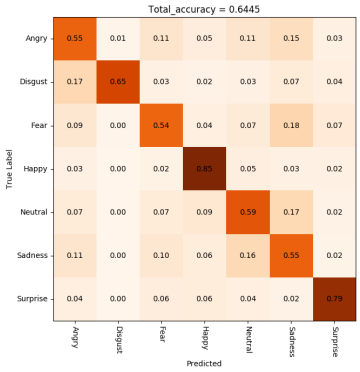


Figure 3.8: FER2013 validation confusion matrix for single identification DenseNet network without weighted-loss approach.

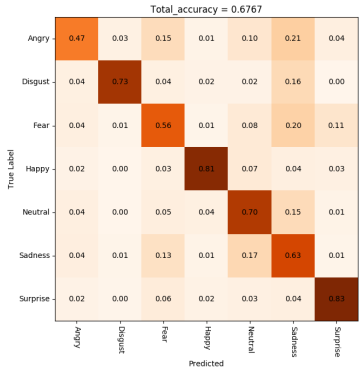


Figure 3.9: FER2013 validation confusion matrix for single identification DenseNet network enhanced with a weighted-loss approach.

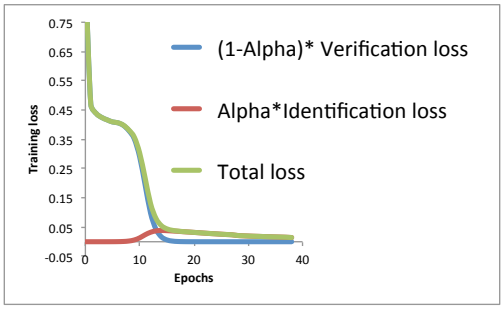


Figure 3.10: Training losses for the proposed architecture for AffectNet dataset.

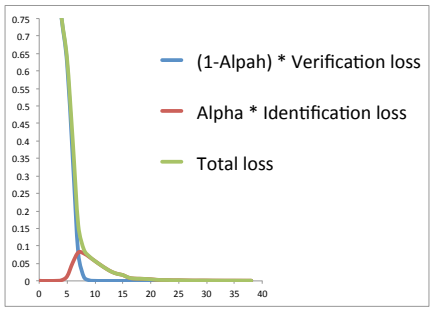


Figure 3.11: Training losses for the proposed architecture for FER2013 dataset.

3.4.3.2 Enhancing the system with verification framework

In the second experiment, we investigate the effect of integrating the verification signal into the deep learned features. Table 3.2 and Figure 3.2c clearly shows that using the verification framework (Loc-SML) along with the identification framework improves the recognition accuracy on the validation set for all datasets. Figures 3.10 and 3.11 shows the training losses for our proposed architecture for AffectNet and FER2013 datasets. The verification loss contributes early, and then after several epochs, the identification loss is activated gradually. Specifically, the verification loss usually begins with a very small value and then begins to decrease until it vanishes while the identification loss dominates the verification loss gradually. As the alpha reaches 1, the identification loss begins to decrease. The shift value in Equation 3.17 is set to 5 except for AffcetNet dataset where it is set to 10. Tuning the shift hyperparameter depends on our observation of the verification loss. A large shift value does not affect the overall accuracy as the verification loss begins to stabilize to a low value at early epochs. These results prove that the verification information included in the learned features of the verification framework helps to reduce the intra-class variation and aids in better generalization for identification task.

Table 3.2: Validation accuracy (%) when using verification framework along with identification framework.

Methods	AffectNet	FER2013	EmotioNet-7	EmotioNet-22
Single Network	61.01 $\pm$0.22	67.89 $\pm$0.20	97.63 $\pm$0.23	69.29 $\pm$0.58
Siamese (Loc-SML)	**64.22 $\pm$0.14**	**72.63$\pm$ 0.29**	**98.41 $\pm$0.40**	**72.07 $\pm$0.42**

3.4.3.3 Investigating our model with local mean vs. global mean and regularized feature

In the third experiment, we investigate using different approaches for the mean calculation of our verification energy loss function mentioned in Equation 3.8. We mentioned in Section 3.2.2.1 that the local means for the batches are computed online while the network is training. The other approach is to calculate the mean for the whole training data after completing one epoch and use that mean instead for feature learning optimization. We also optimize the metric using regularized feature. We compare all these three models with the original SML model in Cao et al. (2013), which does not have the third and fourth terms of mean optimization problem mentioned in our Loc-SML loss function in Equation 3.8.

The results in Table 3.3 is interesting as it shows that using the local mean or regularized feature in our proposed architecture to represent the whole class leads to a better performance than using the global mean or the original SML model. Moreover, using the global mean slows down the training as it needs further computation at the end of each epoch. Therefore, using the local mean or regularized feature: compensates the dependency on the global mean, helps to maximize the distance between samples from each class and the local mean, and thus assists maximizing the inter-class variation.

Table 3.3: Validation accuracy (%) of Loc-SML model with local mean vs. global mean and regularized feature.

Methods	AffectNet	FER2013	EmotioNet-7	EmotioNet-22
Original SML Cao et al. (2013)	56.50 $\pm$0.32	69.16 $\pm$0.21	96.87 $\pm$0.00	67.96 $\pm$0.15
Full-GlobalMean	60.68 $\pm$0.09	68.66 $\pm$0.28	96.45 $\pm$0.97	67.41 $\pm$0.39
Full-LocalMean	63.48 $\pm$0.16	70.49 $\pm$0.24	97.56 $\pm$0.24	69.40 $\pm$0.33
Full-regularized (Loc-SML)	**64.22 $\pm$0.14**	**72.63$\pm$ 0.29**	**98.41 $\pm$0.40**	**72.07 $\pm$0.42**

3.4.3.4 Investigating Partial-1 mining strategy with using all classes vs. close classes for the negative pairs

In the next experiment, we investigate different approaches for **Partial-1** mining strategy to verify which approach gives a better performance. In Section 3.2.1.2, we mentioned **Partial-1** mining strategy for negative pairs that enlarges the distance between the classes that are close to each other. In this experiment, we investigate the power of our mining strategy by comparing it with other possible approaches. The first approach is to use only positive pairs without using any negative pairs, while the second approach is to exploit all the possible negative classes that can be generated from all classes of our datasets. Since the EmotioNet-22 dataset has 22 classes, we do not utilize all possible negative classes approach as it will produce many negative losses, and hence it needs more computational resources and will slow the training process. Table 3.4 shows that both approaches perform less than our mining strategy of considering only the close classes. This experiment verifies that the mining strategy plays a vital role in training the Siamese networks.

Table 3.4: Validation accuracy (%) with different mining strategies for Partial-1 verification framework.

Methods	AffectNet	FER2013	EmotioNet-7	EmotioNet-22
Only Positive Partial-1	61.77 $\pm$0.15	69.18 $\pm$0.38	97.20 $\pm$0.67	66.86 $\pm$0.33
All-classes Partial-1	61.65 $\pm$0.37	68.47 $\pm$0.23	96.18 $\pm$0.49	N/A
Close-classes Partial-1	**62.59 $\pm$0.34**	**70.03 $\pm$0.35**	**97.57 $\pm$0.24**	**70.70 $\pm$0.73**

Table 3.5: Validation accuracy (%) with different mining strategies for Partial-2 verification framework.

Methods	AffectNet	FER2013	EmotioNet-7	EmotioNet-22
Gloabl-mean Partial-2	59.49 $\pm$0.57	67.94 $\pm$0.49	96.00 $\pm$0.48	65.82 $\pm$0.31
Local-mean Partial-2	61.78 $\pm$0.44	70.74 $\pm$0.51	96.87 $\pm$0.73	69.46 $\pm$0.09
Regularized Partial-2	**62.31 $\pm$0.86**	**71.47 $\pm$0.05**	**97.56 $\pm$0.49**	**70.82 $\pm$0.54**

3.4.3.5 Investigating Partial-2 mining strategy with local mean vs. global mean and regularized feature

In the next experiment, we investigate different approaches for **Partial-2** mining strategy to verify which approach gives a better performance. In section 3.2.1.2, we describe **Partial-2** pairing strategy for negative pairs in which we pair the local mean of data with all examples in the data. In this experiment, we investigate replacing the local mean with the global mean and regularized feature approach to notice the power of each one. Table 3.5 shows that the strategy of using the local mean or the regularized feature is better than using global mean.

3.4.3.6 Investigating different models

We investigate using different models which contain either the Euclidean distance (Euclidean model) or Cosine similarity (gCosLA model) along with using the pairing strategy mentioned in Section 3.2.1. More details about the two models are mentioned in Section

Table 3.6: Validation accuracy (%) with different models for verification framework.

Methods	AffectNet	FER2013	EmotioNet-7	EmotioNet-22
Element-wise Euclidean	63.34 ±0.13	71.89±0.11	97.91±0.42	69.54 ±0.40
Element-wise gCosLA	63.37 ±0.31	70.54±0.46	97.91±0.00	70.24±0.40
Pair-wise gCosLA	63.95±0.20	71.70±0.47	97.74±0.24	69.92±0.57
Ours (Loc-SML)	**64.22 ±0.14**	72.63±0.29	98.41±0.40	72.07 ±0.42
Ensemble method	64.00	**73.00**	**98.95**	**72.46**

3.3. We utilize only the Full pairing strategy for these two models since it has been proved to have a better performance than a Partial pairing strategy. Moreover, we enhance the strategy for Cosine similarity model by considering the Pair-wise strategy mentioned in Section 3.2.1.1 with the Element-wise pairing strategy. We do not use the Pair-wise strategy for our model (**Loc-SML** model) as it will consume computational resources with any model that has Euclidean distance function. On the other hand, it is easy to accomplish with any Cosine similarity model by doing proper matrix multiplication. We apply the equation in Qamar and Gaussier (2009) for Cosine similarity model and the equation in McLaughlin et al. (2016) for Euclidean distance model. Table 3.6 shows the results of the two models. It shows how the Pair-wise is better than the Element-wise pairing strategy for the gCosLA model. Moreover, it shows how the Loc-SML model has a better performance than either the Euclidean model or the gCosLA model.

More importantly, we mentioned in Section 3.3 that we will exploit multiple visual features by building an ensemble of different models, in which each model has a different verification framework. Afterward, the best performance due to different pairing strategies for each model will be combined with the best performance for our original model. Table 3.6 shows the results of applying an ensemble on different models. It shows how applying an ensemble method enhances the results for all datasets except the AffectNet dataset in which the result was approximately the same.

3.4.3.7 Comparing our model with state-of-the-art methods

Finally, a comparison with state-of-the-art methods and network architectures for all datasets is shown in Table 3.7. The state-of-the-art work for AffectNet dataset Mollahosseini et al. (2017) is BReG-Net architecture presented in Hasani et al. (2019) along with weighted-loss. As the table shows, our method achieves better accuracy (**64.22%**) on AffectNet. Moreover, we evaluated the performance of several famous architectures such as ResNet, Inception, VGG, MobileNtt, Xception, and InceptionResNet along with weighted-loss on AffectNet, EmotioNet-7, and EmotioNet-22. Table 3.7 shows the state-of-the-art work for other datasets like FER2013. It can be seen that by integrating the verification framework into the identification framework performs better than the state-of-the-art methods. Therefore, the learned features from both frameworks help to produce more discriminative features for facial expression recognition task.

Table 3.7: State of the art comparison (%).

Data	Algorithms	Accuracy(%)
AffectNet	BReG-Net Hasani et al. (2019)	64.04
	ResNet He et al. (2016a)	60.74
	InceptionV2 Szegedy et al. (2016)	59.93
	MobileNet Howard et al. (2017)	61.72
	Vgg16 Simonyan and Zisserman (2014)	61.27
	Xception Chollet (2017)	59.50
	Inception-ResNet Szegedy et al. (2017)	59.59
	Ours	**64.22**
FER2013	Going deeper Mollahosseini et al.	66.40
	FER2013 winner Tang (2013)	71.20
	Multiple deep network learning Yu and Zhang (2015)	72.08
	Adaptive Weighting Xie et al. (2019)	72.67
	Hierarchical committee of DCNNs Kim et al. (2016)	72.72
	Multi-scale CNNs Zhou et al. (2016)	72.82
	Ours	**73.008**
EmotioNet-7	Nearest-mean classifier Du et al. (2014)	92
	Multiclass support vector machine Du et al. (2014)	85.71
	ResNet He et al. (2016a)	93.36
	InceptionV2 Szegedy et al. (2016)	92.19
	MobileNet Howard et al. (2017)	94.14
	Vgg16 Simonyan and Zisserman (2014)	96.88
	Xception Chollet (2017)	93.75
	Inception-ResNet Szegedy et al. (2017)	91.40
	Ours	**98.958**
EmotioNet-22	Nearest-mean classifier Du et al. (2014)	70.3
	Multiclass support vector machine Du et al. (2014)	35.27
	ResNet He et al. (2016a)	68.75
	InceptionV2 Szegedy et al. (2016)	66.02
	MobileNet Howard et al. (2017)	66.60
	Vgg16 Simonyan and Zisserman (2014)	69.53
	Xception Chollet (2017)	62.70
	Inception-ResNet Szegedy et al. (2017)	64.06
	Ours	**72.460**

Chapter 4

Valence and Arousal Estimation in Facial Images using Siamese Neural Networks

While the estimation of emotional valence and arousal in the continuous domain could have several benefits, the current research on automated facial analysis are mostly focused on predicting discrete/categorical emotion. This work introduces an end-to-end algorithm for the valence and arousal estimation using Siamese Neural Networks. The key idea of this approach depends on a multi-task learning (MTL) and transferring learning from one space to another to leverage the vital information across the tasks. The empirical results demonstrate that transferring the learning from the classification space to the regression space enhances the regression task since each expression occupies a representation within a specified range of valence-arousal affect. We introduce several novel strategies for engaging the two spaces within Siamese Neural Networks, ranging from different classification models to different methods for transferring the learning from one space to another. We evaluate the network on AffectNet dataset that contains high variation of facial images in

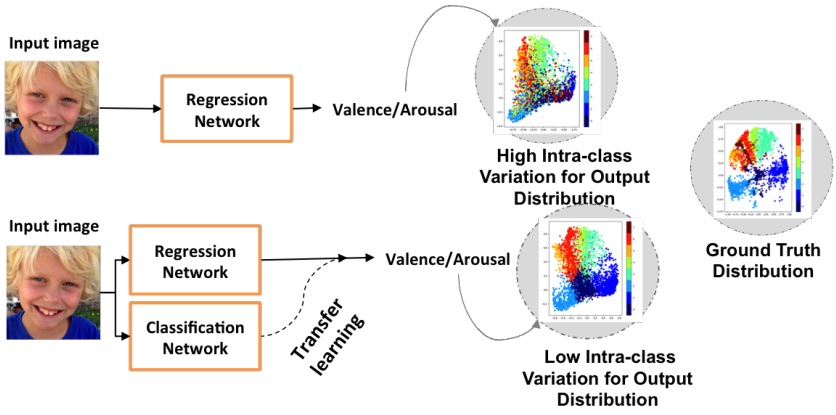


Figure 4.1: An Intuitive illustration of facial expression distribution of AffectNet validation dataset within the 2-dimensional valence-arousal space for single regression network and Siamese networks. It shows that the Siamese distribution is more similar to the ground truth distribution than the single regression network distribution.

the categorical and continuous state where comparable results are achieved compared to state-of-the-art work.

4.1 Overview

Automated facial image analysis has become central in many aspects of our life, such as daily communications with people, medical diagnostics, commercial advertisements, psychological sessions, human-machine interface, and education field. Although automated facial analysis has a long history of studies, it is with the recent achievements in collection of large-scale facial image dataset and progress in deep machine learning methods that the facial affect analysis with high fidelity becomes available for use in many applications.

The emotion analysis can be characterized through three main categories, namely 1) discrete emotion recognition of the basic expressions defined by Ekman Ekman and Friesen (1971) 2) facial Action Units (AUs) detection (e.g., lip tightening and cheek raising) Ekman (1997), 3) continuous emotion estimation in circumplex dimensional model (e.g., valence and arousal) Russell (1980). On a similar line with the continuous emotion, several

works describe the continuous emotion in three dimensions of valence, arousal, and dominance Verma and Tiwary (2017), while other algorithms describe the continuous emotion in four dimensions of arousal, valence, power, and expectancy Kim et al. (2011).

Over the last decade, the majority of the research studies were conducted on discrete emotion recognition or AUs detection. On the other hand, the approaches for modeling the continuous emotion are limited. One of the reasons is the lack of datasets that provide continuous labels. With the recent release of the AffectNet dataset and the availability of continuous annotation, the motivation to investigate the two-dimensional representations using machine learning methods has gained attention. Our main focus in this dissertation is on the continuous emotion defined on Russell's dimensional model of facial affect Russell (1980). In this model, the emotion are revealed in a two-dimensional space, valence and arousal. Valance represents the degree of unpleasant-pleasant (x-axis), while arousal represents the degree of calming-soothing (y-axis) of the expression.

Since the human emotional states of the three preceding tasks of facial affect have different analytical representation, most studies have examined these tasks individually and separately from other tasks Khorrami et al. (2016); Kollias and Zafeiriou (2019a); Li et al. (2017); Lindt et al. (2019). For example, the authors in Lindt et al. (2019) trained a deep generative model that can be used to manipulate the facial images according to continuous two-dimensional emotion labels. They conducted a variety of network architectures to evaluate the valence and arousal of the generated images. The authors in Khorrami et al. (2016) considered combining convolutional neural networks (CNNs) with recurrent neural networks (RNNs) to predict the values of valence and arousal and analyze the effect of each network component on the overall performance.

On the other hand, some studies correlate between the categorical emotion recognition task and continuous emotion estimation task Handrich et al. (2019); Kollias and Zafeiriou (2019b); Kollias et al. (2019a); Siqueira (2018). For example, the authors in Siqueira

(2018) utilized a multi-task learning approach to learn multiple tasks in parallel. They used triple networks that share the low-level representation for emotion recognition and valence/arousal estimation simultaneously. They concluded that training multiple related tasks leads to better generalization and makes the pre-training more efficient. Other studies correlate between the three tasks together, for example, the authors in Kollias et al. (2019a) proposed a multi-task single CNN-based network to jointly learn facial action units, categorical emotion, and dimensional emotion. On a similar line, the authors in Chang et al. (2017) introduced FATAUVA-Net, which is a deep neural framework for facial attribute prediction, facial action unit detection, and valence/arousal estimation. The framework is structured sequentially, taking advantage of the data flow from one task-related layer to the next layer to enhance the performance of other tasks. Therefore, they applied the AU layer and facial attribute layer first and used them to estimate the valence and arousal values in the next layer since both layers are used to reveal the categorical emotion.

In this dissertation, we focus on the joint learning of the valence/arousal regression task and categorical emotion recognition to enhance the regression problem (see Figure 4.1). Our motivation behind this strategy comes from the fact that all our basic expressions are spread in a specific cluster in the circular field space defined by Russell (1980). Accordingly, we believe that the classification space for the categorical emotion recognition has low intra-class variation, which will enhance the high intra-class variation exists in the valence-arousal regression space.

Our major contributions are summarized as follows:

1. Develop a novel deep Siamese Neural Networks architecture that contains a regression framework and a classification framework.

2. Transfer the learning from the classification space to the regression space to enhance the discriminative power of the regression embedding space.

3. Introduce several novel strategies for engaging the two spaces within the Siamese Neural Networks, ranging from different classification models to different methods for transferring the learning from one space to another.

4.2 Related Work

Discrete facial expression recognition algorithms have been actively studied for several decades. Since this dissertation is concerned with the joint learning of continuous emotion and discrete emotion recognition to enhance the regression task, we briefly mention the latest research on continuous emotion regression.

4.2.1 Feature Representation

Learning discriminative features is the first step in any face recognition system. A large variety of algorithms are conducted to extract features from still images or video frames to predict human emotional states. Early works employ handcrafted local features to build feature representations Glodek et al. (2011); He et al. (2015); Meng and Bianchi-Berthouze (2011); Meng et al. (2013); Nicolle et al. (2012); Ramirez et al. (2011) for dimensional predictions of emotional states. For example, the authors in He et al. (2015) concatenated several handcrafted features from audio and video input data and fused the features via Deep Bidirectional Long Short-Term Memory Recurrent Neural Network (DBLSTM-RNN). The authors in Nicolle et al. (2012) extracted dynamic descriptions of signals from the global-, local-face appearance, head movements, and voice. They used a correlation-based measure for the feature selection process and designed a framework for both real-time feature fusion and regression process.

However, a wide range of visual tasks has shown that training using a Deep Convolutional Neural Network (DCNN) can learn more compact and discriminative representa-

tions Girshick et al. (2014); Krizhevsky et al. (2012); Papandreou et al. (2017). To this end, many methods follow deep learning approaches to learn features for the dimensional emotion task Chen et al. (2017); Hasani and Mahoor (2017); Hasani et al. (2019); Khorrami et al. (2016); Kollias and Zafeiriou (2019a); Kollias et al. (2017, 2019b); Li et al. (2017); Lindt et al. (2019). For example, the authors in Hasani et al. (2019) introduced a Bounded Residual Gradient Network(BReG-Net) in which they replaced the identity mapping in deep Residual networks He et al. (2016b) with a differentiable function. The authors in Hasani and Mahoor (2017) presented three different deep networks that are a combination of Inception, ResNet, and LSTM to estimate the values of valence and arousal in the wild. The authors in Mollahosseini et al. (2017) used Alexnet architecture Krizhevsky et al. (2012) to estimate the values of valence and arousal. The authors in Li et al. (2017) presented ensembles of Bi-directional Long Short-Term Memory (Bi-LSTM) networks Graves et al. (2013) to estimate the values of valence and arousal in the wild. While the authors of Khorrami et al. (2016); Kollias et al. (2017, 2019b) considered combining CNNs with RNNs to predict the values of valence and arousal in the wild and analyze the effect of each network component on the overall performance.

Some approaches are different in the sense that they are mixed between the handcrafted features and deep learned features. For example, the authors in Al-Hamadi et al. (2016) extracted the geometrical facial features and then mapped them to a two-dimensional circumplex model of valence-arousal affect. On the other hand, the authors in Chen et al. (2017) compared the performance of engineered features and deep features learned from LSTM-RNN architecture using acoustic, visual, and textual modalities. They applied multi-task learning to predict the valence and arousal values simultaneously.

In this dissertation, we follow our previous work in Hayale et al. in which we used Siamese Neural Networks and introduced our strategy of transferring the the learning from verification space to identification space to enhance the identification space. In this appli-

cation, the two identical networks are one for affect estimation and the other for expression recognition. Both networks are sharing their bottom features extraction and optimized within a multi-task learning approach through two different metrics. Further, we introduce multiple strategies for transferring the learning between the two frameworks based on our objective of reducing the intra-class variation exists in continuous domain of emotional states.

4.2.2 Multi Task Learning

Most of the study over the past few years used single task learning (STL) approaches for continuous emotion estimation Khorrami et al. (2016); Kollias and Zafeiriou (2019a); Li et al. (2017); Lindt et al. (2019). Several approaches moved towards joint learning of valence and arousal to predict their values simultaneously Chen et al. (2017); Kollias et al. (2017, 2019b). Most of other approaches were designed for categorical emotion recognition and re-trained on valence/arousal estimation task by replacing the last layer with one neuron regression layer Barros et al. (2019); Guo et al. (2020); Hasani et al. (2019); Vielzeuf et al. (2019). Others moved towards fine-tuning the trained network on a different task. For example, the authors in Siqueira et al. (2020) presented CNN-based Ensembled with Shared Representation (ESR) model to reduce the cost of ensembling of different decorrelated deep networks. After training the model for facial expression recognition, they fine-tuned the model for valence/arousal prediction by adding two neurons on top of each branch of the ensemble.

On the other hand, some researchers extract different features to enhance the continuous emotion estimation task. For example, the authors in Nicolaou et al. (2011) utilized the features extracted from facial expression, audio cues, and shoulder gesture and then fuses those features using Bidirectional Long Short-Term Memory Neural Networks (BLSTM-

NNs) to enhance the continuous emotion estimation. On a similar line, the authors in Xiaohua et al. (2019) proposed two-levels of attention with a two-stage learning framework which is dedicated to exploit the categorical representation alongside with continuous emotion features and then utilized Bi-directional Recurrent Neural Network(Bi-RNN) to make full use of the relationship features and to improve performance on dimensional emotion estimation.

One interesting approach is FATAUVA-Net that is introduced in Chang et al. (2017), which is a deep neural framework for facial attribute prediction, facial action unit detection, and valence/arousal estimation. The framework is structured sequentially, taking advantage of the flow of data from one task-related layer to the next layer to enhance the performance for other tasks. Therefore, they applied the AU layer and facial attribute layer first and used them to estimate the valence and arousal values in the next layer since both layers are used to reveal the categorical emotion. Additionally, the authors in Mehu and Scherer (2015) showed in their study that the classification of expression into discrete emotion relies on more general information like the dimensional emotion of valence and arousal in the wild.

Lastly, the researchers moved to multi-task learning approaches to jointly learn parallel tasks together. MTL first was introduced in 1999 by Caruana Caruana (1997). The authors demonstrated that by sharing a common representation and transferring the knowledge learned across the tasks can improve the learning of the other related tasks. Since then, more approaches have adopted MTL as the main approach for solving the different problems Hayale et al.; Pan and Yang (2009); Zhang et al. (2013); Zhang and Yang (2017). For example, the authors in Handrich et al. (2019) introduced a single CNN network based on YOLO architecture Redmon and Farhadi (2017) to predict the valence and arousal values alongside with other tasks like the categorical emotion prediction and bounding box detection. The authors in Siqueira (2018) utilized a multi-task learning approach by using triple networks that shared the low-level representation for emotion recognition and

valence/arousal estimation simultaneously. They concluded that training multiple related tasks leads to better generalization and make the pre-training more efficient. The authors in Zhang et al. (2019) introduced PersEmoN, which is a deep Siamese-like network that shares its low-level features representation to investigate the joint learning of apparent personality and emotion analysis from facial images. Both branches of the network are optimized within the multi-task learning framework.

On a similar line, the authors in Kollias and Zafeiriou (2019b) proposed a multi-task CNN-RNN based architecture in which they used the visual and audio data in a Siamese-like architecture. The information from visual and audio streams are fused through two layer GRU in which the joint learning is achieved in terms of facial action units, categorical emotion, and dimensional emotion. The authors in Kollias et al. (2019a) proposed a multi-task single CNN based network to jointly learn facial action units, categorical emotion, and dimensional emotion. The authors in Jang et al. (2019) presented the Face-SSD model for joint learning of multiple face-related tasks like smile detection, facial attributed prediction, and valence/arousal estimation in the wild. Their network has two branches that are sharing the low-level feature representation, one for the face detection and the other for the facial analysis tasks.

Moreover, different than MTL approaches in which both objectives (classification and regression) are weighted equally, we follow our previous work in Hayale et al. in which we proposed a weighting scheme to control the contribution in each framework. For the regression task, we transfer the learning from the classification framework into the regression framework. The classification signal targets the embedding space first at the early training time then it vanishes gradually. Later, the regression signal contributes gradually to the facial affect estimation system. In summary, dynamically modulating the classification signal over the regression signal has the ability to learn effective features representation that can be obtained by decreasing the intra-class variation through the classification signal.

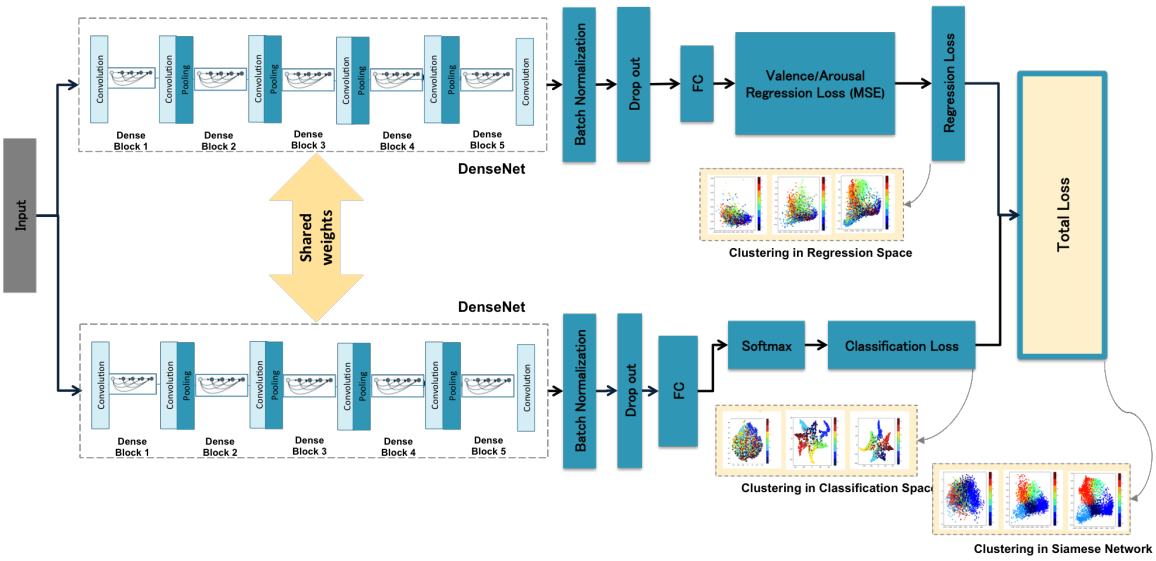


Figure 4.2: Main diagram of our proposed architecture.

4.3 Siamese Neural Networks

Our proposed Siamese architecture, as shown in Figure 4.2 consists of two frameworks. Each framework represents the state-of-art DenseNet network presented in Huang et al. (2017), although other deep network architectures can be utilized. Both frameworks share the weights and parameters within their earlier architecture layers and are responsible for mapping the input images into more discriminative high-level features.

The first network represents the regression framework that is responsible for the valence/arousal estimation task by using the MSE metric as our loss function. The second network represents the classification framework, in which we aim to achieve the multi-class classification using a cross-entropy loss function along with softmax function. In this dissertation we want to explore how discrete emotion representation can be used to enhance the automated valence/arousal estimation. Therefore, our goal for the regression framework is to map the input images into an embedding space that has more discriminative features by using the low intra-class variation existing in the classification framework. Hence, we aim to achieve a multi-task metric learning approach in which we will leverage the perfor-

mance of a single task by combining multiple objective losses and transferring the learning from the learned features of the classification framework to the regression framework.

We train our network using AffectNet Mollahosseini et al. (2017). Then, based on the deep discriminative features, we estimate the values of valence and arousal in the wild. Further details of each component of our Siamese networks are provided in the following subsections.

4.3.1 Regression Framework

The goal of this framework is to predict the emotional states of facial expressions by mapping the input images into 2-dimensional space of the valence-arousal circumplex model. In this framework, the DCNN features obtained from DenseNet network are mapped to fully connected layer of one neuron to represent the valence/arousal continuous value. We employ the MSE as our loss function, which is optimized by taking the squared error of the prediction and the ground truth. Given a feature vector $\hat{y}$ with its associated valence/arousal label y, we can derive the MSE loss as follows:

$$MSE = \frac{1}{K}\sum_{i=1}^{K}(\hat{y} - y)^2 \tag{4.1}$$

where K represents the total number of samples. The main evaluation metrics that are used to evaluate our regression framework performance are Root Mean Squar Error (RMSE), Pearson's Correlation Coefficient (CC), Concordance Correlation Coefficient (CCC), and Sign Agreement (SAGR). The RMSE represents a common comparative metric for the continuous domain and a small value is desired. It can be defined as follows:

$$RMSE = \sqrt{\frac{1}{K}\sum_{i=1}^{K}(\hat{y} - y)^2} \tag{4.2}$$

where $\hat{y}$ represents the prediction value with y as its associated expression label, and K represents the total number of samples. The RMSE-based evaluation methods weights the outliers heavily Bermejo and Cabestany (2001) and do not consider any structural covariance information that relates the changes in $\hat{y}$ and y Nicolaou et al. (2011). The CC overcomes this problem by including the covariance between these two values as follows:

$$COR(\hat{y}, y) = \frac{COV\{\hat{y}, y\}}{\sigma_{\hat{y}}\sigma_y} = \frac{\frac{1}{K}\sum_{i=1}^{K}(\hat{y} - \mu_{\hat{y}})(y - \mu_y)}{\sigma_{\hat{y}}\sigma_y} \tag{4.3}$$

where COV represents the covariance between $\hat{y}$ and y, σ represents the standard deviation, and μ stands for the mean value. The CCC is another metric which combines the CC with the square difference of $\mu_{\hat{y}}$ and μ_y. It is defined as follows:

$$CCC = \frac{2p\sigma_{\hat{y}}\sigma_y}{\sigma_{\hat{y}}^2 + \sigma_y^2 + (\mu_{\hat{y}} - \mu_y)^2} \tag{4.4}$$

where p represents the Pearson's Correlation Coefficient (CC). Another important metric is SAGR which is introduced in Nicolaou et al. (2011) to evaluate the performance of the regression model depending on the sign agreement between the valence and arousal values. It is defined as follows:

$$SAGR = \frac{1}{K}\sum_{i=1}^{K}\delta(sign(\hat{y}), sign(y)) \tag{4.5}$$

where δ represents the Kronecker delta function that is defined as:

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \tag{4.6}$$

4.3.2 Classification Framework

The second branch in our network is the classification framework, in which we use a second DenseNet network to map the images into high dimensional feature space by using softmax layer to get the probability distribution of each emotion class over all the categorical classes. The probability distribution of one class can be defined as follows:

$$P_i = P(y = i|f) = \frac{\exp^{W_i f}}{\sum_k \exp^{W_k f}} \tag{4.7}$$

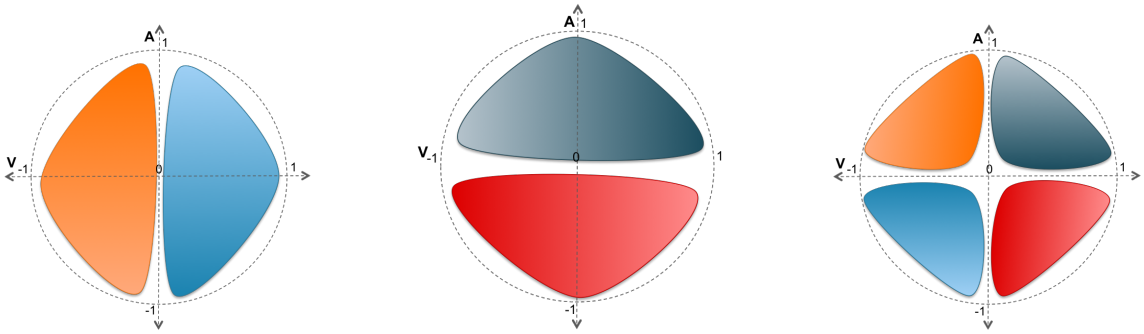
where W_i refers to the i$^{\text{th}}$ column of softmax weight matrix. Accordingly, the network of the classification framework is trained to minimize the cross-entropy loss as defined below:

$$loss_c = -\sum_{i=1}^{K} y \log P_i \tag{4.8}$$

where $y = 0$ stands for all classes except for the target class, and K represents the total number of examples in the dataset.

For this framework, we investigate four types of models according to the number of classes that can be constructed throughout the system. The first model is the categorical emotion classification model in which the objective is to map the facial images into one of the eights facial emotion classes. The second model classifies the image into four classes based on the estimation values of valence and arousal in the circumplex dimensional model as shown below:

$$Model2_{out} = \begin{cases} class_0 & V \geq 0 \quad \& \quad A \geq 0 \\ class_1 & V < 0 \quad \& \quad A \geq 0 \\ class_2 & V < 0 \quad \& \quad A < 0 \\ class_3 & V \geq 0 \quad \& \quad A < 0 \end{cases} \tag{4.9}$$



(a) Valence Classification (b) Arousal Classification (c) Valence-Arousal Classification

Figure 4.3: Different classification models.

Table 4.1: Different classification models.

Model-Name	Model-Description	FC-Neurons
Model 1	Facial Expressions Classification model	8
Model 2	Discrete Valence-Arousal Classification	4
Model 3	Discrete Valence Classification	2
Model 4	Discrete Valence Classification	2

While, the third and fourth model classifies the image into two classes based on either valence or arousal value as shown below:

$$Model3_{out} = \begin{cases} class_0 & V \geq 0 \\ class_1 & V < 0 \end{cases} \tag{4.10}$$

$$Model4_{out} = \begin{cases} class_0 & A \geq 0 \\ class_1 & A < 0 \end{cases} \tag{4.11}$$

More details about the models are in Table 4.1 and Figure 4.3.

4.3.3 Joint Regression-Classification Learning

The learned features of both frameworks are revealed to have different characteristics. The 2-dimensional features in the regression space emphasize valence and arousal variation, while the learned features in the classification space emphasize the inter- and intra-class variation that exist between the categorical classes. In order to construct an effective facial affect estimation system that benefits from both spaces, we join both frameworks on the top by using a controlling function that controls the contribution of each framework. More precisely, we jointly optimize both frameworks' costs to the extent that we can enhance system performance. We observed that the initial activation of the classification framework enhances the clustering of valence/arousal learned features of the training dataset. Consequently, vanishing the classification signal from the system along with gradually integrating the regression signal enhances the overall performance of the system. Hence, we can derive the overall Siamese loss function as follows:

$$Total_{loss} = (1 - \alpha).loss_c + \alpha.loss_r \qquad \alpha : 0 \rightarrow 1 \tag{4.12}$$

$$\alpha = \frac{1}{\exp \frac{epoch_{num} - shift}{10}} \tag{4.13}$$

where $loss_r$ stands for regression loss and represents the loss function for valence/arousal regression model discussed in Section 4.3.1. On the other hand, $loss_c$ stands for classification loss and represent the loss function for one of the four models discussed in Section 4.3.2. According to the above equation, the $shift$ factor acts as a shifting epoch in which the Siamese networks has an equal contribution from $loss_r$ and $loss_c$ to $Total_{loss}$. More precisely, after this shifting epoch, the classification signal begins to vanish, and the regression signal begins to integrate into the system.

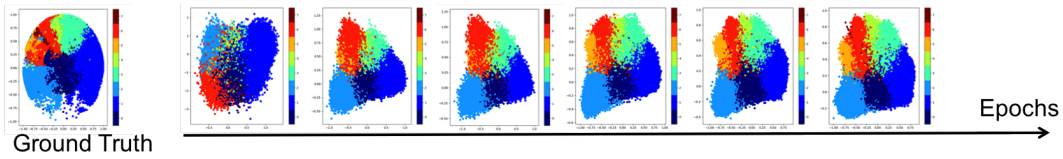
Ultimately, after constructing the Siamese networks by using one of the four classification models with the regression model, we employ three strategies for learning Siamese networks. The first strategy involves learning Siamese networks from scratch without loading the weights from previously learned classification or regression single network. The second strategy requires loading the weights from the trained single classification network. Finally, the third strategy requires loading the weights from a trained single regression network. A full comparison between all the models was mentioned in Section 4.4.3.2.

Accordingly, with the three learning strategies and the four classification models, we will have 12 types of Siamese networks for Valance/Arousal estimation in the wild, as shown in Table 4.2.

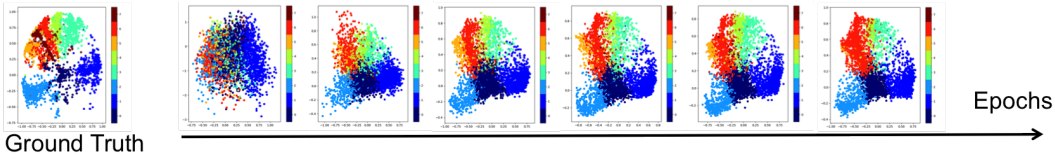
Later on, we exploit four strategies of engaging the two frameworks into the Siamese networks by activation and deactivating the α parameter in Equation 4.12. More precisely, by discarding the α from either signals means we will have a steady signal into the system. Our first strategy includes keeping both signals steady in the system and notice the performance for the valence/arousal estimation, while the second strategy requires gradually engaging the regression signal into the system while the classification signal is steady in the system. The third strategy requires keeping the regression signal steady and vanishing the classification signal after a while. The fourth strategy require vanishing the classification signal while engaging the regression signal gradually in the system. A full comparison between all the models was mentioned in Section 4.4.3.3.

4.4 Experiments and Results

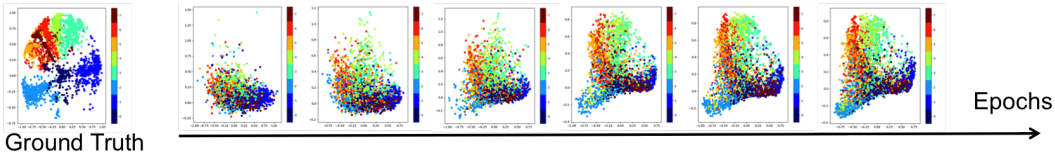
This section presents the experiments and results achieved by using our proposed architecture for facial affect analysis on AffectNet dataset. We introduce the dataset, implementation details for our network, and the experiments in the following subsections.



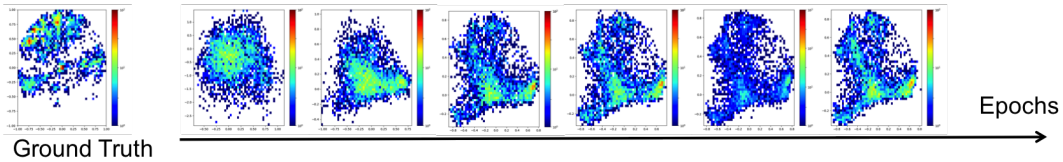
(a) Training valence-arousal distribution for Siamese networks



(b) Validation valence-arousal distribution for Siamese networks



(c) Validation valence-arousal distribution for single regression network



(d) Validation valence-arousal histogram for Siamese networks

Figure 4.4: AffecNet dataset distribution for single and Siamese networks within the valence-arousal space where the indices 0 through 7 represent Neutral, Happy, Sadness, Surprise, Fear, Disgust, Angry and Contempt classes, respectively. The distribution for Siamese networks in (b) reveals small intra-class variation that appear from the clustering of examples belong to the same class to each other. On the other hand, there is no distinctive distance between different classes for a single regression network in (c), which refers to high intra-class variation. We show the histogram within the valence-arousal space for Siamese networks in (d).

Table 4.2: Different Siamese networks.

Siamese	Network1 (Regression)	Network2 (Classification)	Load-weight
SNN_1	V/A model	Model1	None
SNN_2	V/A model	Model1	Model1
SNN_3	V/A model	Model1	V/A model
SNN_4	V/A model	Model2	None
SNN_5	V/A model	Model2	Model2
SNN_6	V/A model	Model2	V/A model
SNN_7	V/A model	Model3	None
SNN_8	V/A model	Model3	Model3
SNN_9	V/A model	Model3	V/A model
SNN_{10}	V/A model	Model4	None
SNN_{11}	V/A model	Model4	Model4
SNN_{12}	V/A model	Model4	V/A model

4.4.1 Datasets

The **AffectNet** dataset Mollahosseini et al. (2017) is one of the largest datasets provided for facial affect analysis. It has about 1M facial images that are provided by dimensional and categorical representations. The basic facial expressions are annotated manually after the images are generated by querying several search engines. The evaluation protocol is achieved on the validation set since the test set is not released.

4.4.2 Implementation Details

DenseNet architecture presented in Huang et al. (2017) is used as our deep CNNs baseline, though other networks can be utilized. We use the bounding box that is available with AffectNet dataset files in order to crop the faces from the images. The faces are resized to 106×106 pixels. Additionally, we perform landmark-based face alignment and per-image

standardization that linearly scales each image to have zero mean and variance equal to one.

Seven types of augmentations, such as flip, brightness, contrast, rotation, hue, cropping, and saturation, are applied to create more training samples. The network is trained using a batch size of 128. We use Adam optimizer Kingma and Ba (2014) as an adaptive learning rate optimization algorithm for training our deep neural network. The baseline learning rate is set to 0.001 and decreased by a factor of 0.1 when the metric stops improving after every ten epochs.

To get the best computational performance, we use TensorFlow as the most popular and efficient machine learning tool for training our network. Keras is also used as a high-level neural networks API (Application Program Interface) that wraps a sequence of complicated underlying TensorFlow operations. Moreover, we run our experiments on one NVIDIA 1080 Ti GPU (Graphics Processing Unit) as underlying computing devices.

4.4.3 Experiments

We present the results of different experiments on the AffectNet dataset to demonstrate the effectiveness of our purposed model. We finally compare with state-of-the-art methods.

4.4.3.1 Integrating the single network vs. Siamese networks

In this experiment, we investigated the effect of integrating the classification signal into the learned features of the regression framework. Table 4.3 clearly shows that the performance of using the classification framework, along with the regression framework, improves the valence and arousal prediction on the validation set for the regression framework. Figure 4.5 shows the training losses for our proposed architecture outputs where the classification loss contributes early, and then after epoch five, the regression loss is

Table 4.3: Performances of valence and arousal prediction on validation set with Siamese networks vs. single regression network.

Affect	Net	RMSE	CC	CCC	SAGR
V	Single	0.4027 ±0.0005	0.6168 ±0.0031	0.5795 ±0.0012	0.7517 ±0.0004
	SNN	**0.3889 ±0.0007**	**0.6381 ±0.0010**	**0.6058 ±0.0010**	**0.7622 ±0.0019**
A	Single	0.353 ±.0000	0.5232 ±0.0001	0.4686 ±0.0003	0.7819 ±0.0001
	SNN	**0.3444 ±0.0002**	**0.5606 ±0.0004**	**0.4973 ±0.0063**	**0.8006 ±0.0004**

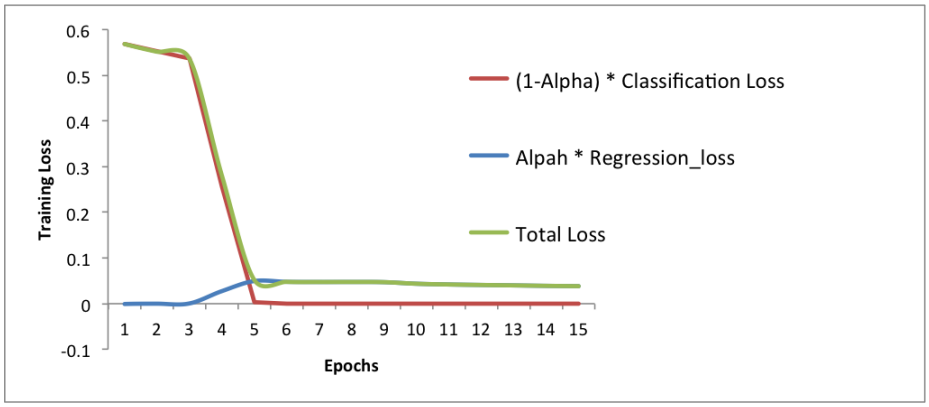


Figure 4.5: Training losses for Siamese networks.

activated gradually. These results prove that the classification information included in the learned features of the classification framework helps to decrease the intra-class variation in the regression framework.

4.4.3.2 Investigating different classification frameworks along with different learning strategies

In this experiment, we investigate the effect of integrating different models for classification framework into Siamese networks (see Table 4.1). More precisely, we want to examine the effect of clustering the embedding features into several classes on the overall valence/arousal estimation of the regression framework. Moreover, we will investigate the three learning strategies of loading the weights into Siamese networks mentioned in Sec-

Table 4.4: Performance of valence prediction on validation set with different Siamese networks.

Siamese	RMSE	CC	CCC	SAGR
SNN_1	0.3980 $\pm$0.0002	0.6131 $\pm$0.0013	0.5671 $\pm$0.0013	0.7418 $\pm$0.00080
SNN_2	**0.3889 $\pm$0.0007**	**0.6381 $\pm$0.0010**	**0.6058 $\pm$0.0010**	**0.7622 $\pm$0.00190**
SNN_3	0.4077 $\pm$0.0004	0.6191 $\pm$0.0014	0.5871 $\pm$0.0012	0.7495 $\pm$0.00110
SNN_4	0.4236 $\pm$0.0008	0.5782 $\pm$0.0007	0.5475 $\pm$0.0008	0.7347 $\pm$0.00110
SNN_5	0.4132 $\pm$0.0008	0.6104 $\pm$0.0012	0.5649 $\pm$0.0015	0.7431 $\pm$0.00120
SNN_6	0.4039 $\pm$0.0003	0.6163 $\pm$0.0015	0.5824 $\pm$0.0014	0.7521 $\pm$0.00050
SNN_7	0.4167 $\pm$0.0007	0.5911 $\pm$0.0012	0.5386 $\pm$0.0009	0.7312 $\pm$0.00054
SNN_8	0.4088 $\pm$0.00016	0.6086 $\pm$0.0009	0.5688 $\pm$0.00072	0.7399 $\pm$0.00041
SNN_9	0.4056 $\pm$0.00088	0.6151 $\pm$0.00043	0.5765 $\pm$0.00047	0.7432 $\pm$0.00083
SNN_{10}	0.4272 $\pm$0.00048	0.5886 $\pm$0.0016	0.532 $\pm$0.00111	0.7232 $\pm$0.00064
SNN_{11}	0.4093 $\pm$0.00047	0.6010 $\pm$0.00117	0.5536 $\pm$0.00085	0.7379 $\pm$0.00120
SNN_{12}	0.4003 $\pm$0.00003	0.6091 $\pm$0.00154	0.5866 $\pm$0.00152	0.7513 $\pm$0.00050

tion 4.3.3. Table 4.4 and Table 4.5 shows that using the FER classification and clustering the embedding features into more number of classes (8 classes) results in better features representation than clustering the embedding features into four or two classes. This behavior is because each expression is concentrated over a specific range of valence-arousal dimensional model. However, there is no significant difference between the performances when using four- or two-class model. It also shows that loading weights of the eight class classification model into Siamese networks is better than learning from scratch or loading weights of the regression model into Siamese networks. On the other hand, loading the weights of four- or two-class model into Siamese networks performs less than loading the regression model into Siamese networks. This behavior is because the trained network of four- or two-class classification framework does not have a right cluster for the Neutral class, which occupies the center of 2-dimensional valence-arousal embedding space.

Table 4.5: Performance of arousal prediction on validation set with different Siamese networks.

Siamese	RMSE	CC	CCC	SAGR
SNN_1	0.3608 $\pm$0.00042	0.5495 $\pm$0.00105	0.4455 $\pm$0.00148	0.7854 $\pm$0.00130
SNN_2	**0.3444 $\pm$0.00021**	**0.5606 $\pm$0.00045**	**0.4973 $\pm$0.00632**	**0.8006 $\pm$0.00046**
SNN_3	0.3582 $\pm$0.00042	0.5533 $\pm$0.00101	0.4594 $\pm$0.00098	0.7851 $\pm$0.00059
SNN_4	0.3695 $\pm$0.0002	0.5163 $\pm$0.0009	0.4171 $\pm$0.00113	0.7585 $\pm$0.00105
SNN_5	0.3591 $\pm$0.00084	0.5285 $\pm$0.0029	0.4439 $\pm$0.00282	0.7944 $\pm$0.00163
SNN_6	0.357 $\pm$0.00055	0.5434 $\pm$0.00121	0.4721 $\pm$0.00184	0.7935 $\pm$0.00041
SNN_7	0.3568 $\pm$0.0007	0.5177 $\pm$0.00225	0.432 $\pm$0.00218	0.7829 $\pm$0.00019
SNN_8	0.355 $\pm$0.00047	0.5400 $\pm$0.00118	0.4577 $\pm$0.00184	0.7879 $\pm$0.00060
SNN_9	0.355 $\pm$0.001	0.5436 $\pm$0.00155	0.4929 $\pm$0.00148	0.8024 $\pm$0.00086
SNN_{10}	0.3581 $\pm$0.00021	0.5205 $\pm$0.00053	0.4296 $\pm$0.00053	0.7651 $\pm$0.00071
SNN_{11}	0.3569 $\pm$0.00052	0.5404 $\pm$0.00158	0.4755 $\pm$0.00161	0.7703 $\pm$0.00075
SNN_{12}	0.3544 $\pm$0.00057	0.5411 $\pm$0.00038	0.4906 $\pm$0.00048	0.7882 $\pm$0.00141

4.4.3.3 Investigating different strategies for coupling the two frameworks

In this experiment, we investigate the best way of engaging both signals into Siamese networks. As we mentioned in Section 4.3.3, we have four main strategies for engaging both signals into Siamese networks. Table 4.6 shows that using the fourth strategy of gradually engaging the regression signal into Siamese networks while vanishing the classification signal is better than other strategies. However, it also shows that keeping the regression signal steady or engaging it gradually, does not have a significant effect on the performance while vanishing the classification signal (third and fourth strategy) have a significant effect compared with keeping this signal steady (first and second strategy) during training.

4.4.3.4 Comparing our model with state-of-the-art methods

We finally compared with the state-of-the-art methods for this dataset, as shown in Table 4.7. In the state-of-the-art work Hasani et al. (2019), they used the BReG network, which is

Table 4.6: Performances of valence and arousal prediction on validation set with different joining strategies.

Effect	Strategy	RMSE	CC	CCC	SAGR
V	Strategy_1	0.409 $\pm$0.0033	0.6044 $\pm$0.0026	0.5603 $\pm$0.0025	0.7433 $\pm$0.0019
	Strategy_2	0.4061 $\pm$0.0006	0.6135 $\pm$0.0011	0.577 $\pm$0.0012	0.7438 $\pm$0.0010
	Strategy_3	0.3987 $\pm$.0000	0.6271 $\pm$0.0004	0.597 $\pm$0.0013	0.7477 $\pm$0.0014
	Strategy_4	**0.3889 $\pm$0.0007**	**0.6381 $\pm$0.0010**	**0.6058 $\pm$0.0010**	**0.7622 $\pm$0.0019**
A	Strategy_1	0.3513 $\pm$0.0105	0.541 $\pm$0.0008	0.4358 $\pm$0.0011	0.7890 $\pm$0.0007
	Strategy_2	0.3515 $\pm$0.0006	0.54 $\pm$0.0143	0.4726 $\pm$0.0013	0.7892 $\pm$0.0008
	Strategy_3	0.3463 $\pm$0.00039	0.5644 $\pm$0.0003	0.4843 $\pm$0.0005	0.7913 $\pm$0.0007
	Strategy_4	**0.3444 $\pm$0.0002**	**0.5606 $\pm$0.0004**	**0.4973 $\pm$0.0063**	**0.8006 $\pm$0.0004**

a robust architecture that replaced the identity mapping in deep Residual networks He et al. (2016b) with a differentiable function. As the table shows, our approach of embedding the classification signal achieves a better performance than other state of the art approaches and less performance than BReG network. We did not enhance the DenseNet architecture used in our Siamese networks. Therefore, we believe that building the Siamese networks from a robust architecture like BReG network can boost the performance. It can be seen that the combination of regression signal and facial emotion classification signal plays a vital role in the deep feature representation, and hence the estimation of error for the regression system decreased.

Table 4.7: State of the art comparison.

Affect	Algorithms	RMSE	CC	CCC	SAGR
V	Jang Jang et al. (2019)	0.4406	0.66	0.60	0.74
	Lindt Lindt et al. (2019)	0.450	N/A	0.484	0.676
	Guo Guo et al. (2020)	0.39	0.61	0.59	0.76
	SVR Drucker et al. (1997)	0.55	0.35	0.57	0.30
	Hasani Hasani et al. (2019)	**0.2597**	**0.66**	**0.66**	0.73
	Our	0.3889	0.6381	0.6058	**0.7622**
A	Jang Jang et al. (2019)	0.3937	0.54	0.4665	0.7129
	Lindt Lindt et al. (2019)	0.411	N/A	00.405	0.708
	Guo Guo et al. (2020)	0.37	0.55	0.48	0.76
	SVR Drucker et al. (1997)	0.42	0.31	0.68	0.18
	Hasani Hasani et al. (2019)	**0.3067**	**0.84**	**0.82**	**0.84**
	Our	0.3444	0.5606	0.4973	0.8006

Chapter 5

Image Matching with Siamese Neural Networks

5.1 Overview

The ability to find images that are similar to a query image is a fundamental issue of several computer vision problems like, image retrieval, object verification, wide-baseline matching, and duplicate product detection.

Over the last decade, several algorithms have been proposed to improve the accuracy and performance of image matching applications. In general, these algorithms can be divided into two general categories. The first category involves extracting hand-crafted features and learn the similarity metric on top of them to predict whether the pair of images belong to similar class, or they represent different classes Boureau et al. (2010); Cao et al. (2011); Chechik et al. (2010); Frome et al. (2007); Taylor et al. (2011); Wang et al. (2014b); Wengert et al. (2011). For example, the authors in Frome et al. (2007), learned fine-grained image similarity ranking model on top of the hand-crafted features by learning a distance function for each input image as a combination of distances between patch-based visual

features. The performance of these methods depends on the representation power of the hand-crafted features. On the other hand, the second category is based on deep learning models, which involve extracting the deep features from neural networks. The current state of the art shows that deep learning-based approaches have successfully provided great potential for more effective image similarity models than hand-crafted features.

However, the researchers in deep learning-based models followed different approaches to learn a feature representation so that similar inputs are mapped close to each other in the feature space, and dissimilar inputs are mapped far from each other. The first approach Babenko et al. (2014); Chandrasekhar et al. (2016) is to utilize a pre-trained CNN on image classification problem in image retrieval application. At query time, the feature is extracted for the query image and compared with the features of all dataset by using the Euclidean distance metric and associating the proper class label for query image depending on the closest distance.

The second approach is to learn the similarity directly from image pairs Hadsell et al. (2006); Melekhov et al. (2016); Taylor et al. (2011); Wang et al. (2014a); Wu et al. (2013). In this approach, the researchers used Siamese networks and utilized the similarity labels to learn a feature representation so that similar input pairs are mapped close to each other and dissimilar input pairs are mapped far from each other in the feature space. This approach can be further divided into two further approaches depending on how the model will estimate the similarity score from the pair of images. The first approach Hadsell et al. (2006); Taylor et al. (2011); Wang et al. (2014a) is based on metric distance function to estimate the similarity score directly from a pair of feature vectors, while the other approach determines the similarity score by using a non-metric similarity network. In this approach, either they consider the two images of an input pair as a one 2-channel image Zagoruyko and Komodakis (2015), which is directly fed to the first layer of the network, or the features of both input images are concatenated to create one feature vector for pair of inputs

Han et al. (2015); Tian et al. (2017). For both approaches, the output feature is then given as input to a top module that consists of a fully connected linear decision layer with one (matching/non-matchin) output.

Along a different line, there are few researches on integrating the identification network with any of the verification approaches mentioned above. For example, the author in Sun et al. (2014) presented the idea of using two identification networks and learn a similarity metric on the top of these networks. Therefore, the loss function involves two parts, one to optimize the identification network and the other part to learn the similarity metric directly from the pair of images. The identification and verification losses are weighted by a hyper-parameter. Motivated by this approach, this dissertation tries to investigate this approach with a difference in controlling the way of integrating both frameworks. Instead of learning a hyperparameter for weighting these two losses, this dissertation involves transferring the learning from the identification framework to the verification framework by which the identification signal contributes at the early training time. Later, this signal vanishes while the verification signal is involved gradually in the Siamese architecture.

5.2 Related Work

In the last years, several methods have been proposed to represent visual information in images by a set of hand-crafted features such as SIFT Lowe (2004). For a given a query image, images in the dataset are ranked according to their visual similarity to the query image. In Bow model Sivic and Zisserman (2003), features such as SIFT Lowe (2004) are extracted from pair of images, then the similarity of the pair can be computed as a dot product or histogram intersection of their weighted histograms. Along a similar line, authors in Cao and Snavely (2012) learned to predict the matching and non-matching input pairs with small amounts of training data using discriminative learning of BoW model.

Since the substantial advancement in deep learning using the convolutional neural network (CNN), the features obtained from CNN have become the new state of the art in image matching problem. At first, several approaches Babenko et al. (2014); Chandrasekhar et al. (2016) proposed the idea of utilizing a pre-trained network for image classification problems in image matching. On the other hand, other approaches learn a CNN directly for the matching task Hadsell et al. (2006); Melekhov et al. (2016); Taylor et al. (2011); Wang et al. (2014a); Wu et al. (2013). The mainstream architecture in these approaches is the Siamese and triplet networks. The similarity labels of the pairs are utilized by these networks to map the pairs into a feature space where similar pairs are close to each other, and dissimilar pairs are far from each other in the feature space. Moreover, most of these approaches Hadsell et al. (2006); Taylor et al. (2011); Wang et al. (2014a) used a regular distance function like Euclidean distance on top of the image pair representation to learn the similarity metric. However, there are several approaches used non-metric similarity function on top of the image representation Han et al. (2015); Tian et al. (2017); Zagoruyko and Komodakis (2015). They directly applied the output of the model as a similarity estimation to rank images accordingly without using any similarity metric. The last fully-connected layer acts as a decision layer with one (0/1) output, reflecting the pairs being a similar or dissimilar.

In this dissertation, we depend on the Siamese networks to learn directly from the pair of images, but we enhanced it with the identification signal. Accurately, we transfer the learning from the identification network into the verification network while we are jointly training the full architecture. The Siamese networks automatically learns a representation of the input pairs based on the multiple objective loss function which integrate the identification loss function with the verification loss function. The identification network will enhance the model performance since it helps to increase the inter-class variation.

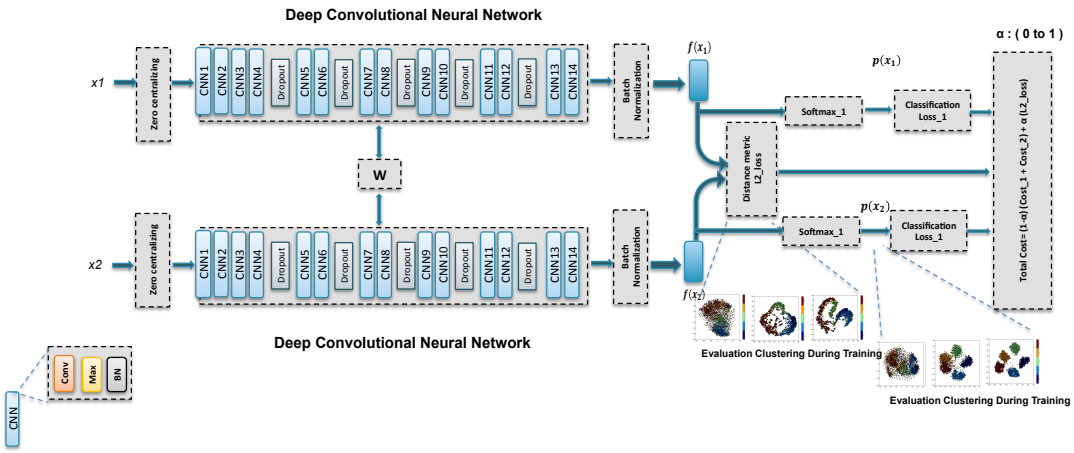


Figure 5.1: Main diagram of our proposed architecture.

5.3 Model Definition

A diagram of our proposed architecture is shown in Figure 5.1. In our architecture a pair of images is first processed by a Siamese networks. In this network, two deep convolutional neural networks consisting of 14 layers with identical parameters produce a feature vector representation for each image. Given a pair of images, the Siamese architecture learns to map those inputs to a feature space where similar inputs are close and dissimilar inputs are separated by a margin. In this dissertation, we show that these features can be more discriminative by using both identification and verification signals as supervision signals. In addition to that, we can control these two signals to enable transferring the learning from the high dimensional features within the identification space (in which all the classes are clustered very well) into the verification space in order to enhance the verification task objective. Hence, we trained our network on the Cifar-100 dataset and derive the joint identification-verification metric using the DCNN features. Then, given a pair of input images, the Siamese model determine if they are similar or not based on their DCNN features and the learned metric. Further details of each component of our model are introduced in the following subsections.

5.3.1 Data Mining

In our Siamese Neural Networks, we trained a deep neural network to learn a set of hierarchical non-linear transformations to project pair of images into a feature space, under which the distance of each positive pair is reduced, and the distance of each negative pair is enlarged. Therefore, the proposed loss function accepts a pair of positive and negative features to optimize the similarity or the distance between them.

Consider Pos, and Neg are two sets of all possible positive and negative pairs that can be generated from the training images that have M categories. Assuming each class has N number of images, then the total number of pairs are:

$$Pos = \sum_{i=1}^{M} N_i * (N_i - 1) \tag{5.1}$$

$$Neg = \sum_{i=1}^{M} \sum_{j=1}^{M} N_i * N_j \qquad i \neq j \tag{5.2}$$

As we stated in an earlier chapter, the mining strategy plays an essential role in Siamese networks training as it relies on the best representative pairs to produce gradient with a sufficiently large magnitude. For this study, we depend on an offline pairing strategy in which all the negative and positive pairs are prepared in advance. This strategy, compared to the online pairing strategy that is discussed in Chapter 3, is very time consuming and difficult to accomplish with a large-scale dataset. With a large volume of data pairs, it will result in a slow training convergence. Therefore, we design the pairs in such a way that each class will have an equal number of positive and negative pairs, which is equal to the number of examples in each class. For example, in Cifar-100 training dataset, each class has 400 examples. Ultimately, we will have 400 positive pairs and 400 negative pairs. For the positive pairs, we pair each example with a random example from the same class. On

the other hand, for the negative pairs, we pair each example with a random example from other classes. Additionally, for negative pairs, we carefully design them to involve all other classes. Specifically, each class will be paired with all other classes and with the same number of negative examples from each class.

5.3.2 Deep Features Representation

The proposed architecture (i.e., Siamese) has two identical networks sharing the same parameters. For each network, we use the same architecture presented in Clevert et al. (2015). Further details about this architecture is given in Table 5.1. The network includes 14 layers, followed by one fully connected layer. Each individual layer consists of a convolutional layer, a max-pooling layer, and a batch normalization layer. For network regularization we use five drop-out layers distributed after several layers with the following drop-out rates [0.1, 0.2, 0.3, 0.4, 0.5]. The max-pooling layer is not activated for some layers, but for simplicity, we keep it in the graph and the table with a stride equal to one. Each convolutional layer is followed by an exponential linear unit (ELU) Clevert et al. (2015). The extracted features are normalized to unit length by using the local response normalization Krizhevsky et al. (2012) method. This final step helps us to set the margin to a proper value in the training when trying to separate the impostor pairs by the specified margin.

When a Siamese model is presented with a pair of images, the networks map the pair of input images to a pair of feature vectors. These features are learned with two supervisory signals. The first signal is the verification loss signal in which pair of features are compared using Euclidean distance. During training, the similar and dissimilar input pairs are presented on Siamese networks which learn to map those pairs from the same class to feature vectors that are close to each other, and map the pairs from different classes to feature vectors that are separated by a margin.

Table 5.1: The architecture details of DCNN.

Layer	Sublayer	Filter size	Input channel	Output channel	Strides (batch,height, width,depth)
	Conv1	(3, 3)	3	384	(1, 1, 1, 1)
Lyaer1	Max1	(2, 2)	1	1	(1, 2, 2, 1)
	Conv2	(1,1)	384	384	(1, 1, 1, 1)
Lyaer2	Max2	(1, 1),	1	1	(1, 1, 1, 1)
	Conv3	(2,2)	384	384	(1, 1, 1, 1)
Lyaer3	Max3	(1, 1),	1	1	(1, 1, 1, 1)
	Conv4	(2, 2)	384	460	(1, 1, 1, 1)
Lyaer4	Max4	(2, 2)	1	1	(1, 2, 2, 1)
	Conv5	(1, 1)	460	460	(1, 1, 1, 1)
Lyaer5	Max5	(1, 1)	1	1	(1, 1, 1, 1)
	Conv6	(2, 2)	460	786	(1, 1, 1, 1)
Lyaer6	Max6	(2, 2)	1	1	(1, 2, 2, 1)
	Conv7	(1, 1)	786	786	(1, 1, 1, 1)
Lyaer7	Max7	(1, 1)	1	1	(1, 1, 1, 1)
	Conv8	(2, 2)	786	896	(1, 1, 1, 1)
Lyaer8	Max8	(2, 2)	1	1	(1, 2, 2, 1)
	Conv9	(3, 3)	896	896	(1, 1, 1, 1)
Lyaer9	Max9	(1, 1)	1	1	(1, 1, 1, 1)
	Conv10	(2, 2)	896	1024	(1, 1, 1, 1)
Lyaer10	Max10	(2, 2)	1	1	(1, 2, 2, 1)
	Conv11	(1, 1)	1024	1024	(1, 1, 1, 1)
Lyaer11	Max11	(1, 1)	1	1	(1, 1, 1, 1)
	Conv12	(2, 2)	1024	1152	(1, 1, 1, 1)
Lyaer12	Max12	(2, 2)	1	1	(1, 2, 2, 1)
	Conv13	(1, 1)	1152	1152	(1, 1, 1, 1)
Lyaer13	Max13	(2, 2)	1	1	(1, 2, 2, 1)
	Conv14	(1, 1)	1152	256	(1, 1, 1, 1)
Lyaer14	Max14	(2, 2)	1	1	(1, 2, 2, 1)

Given a pair of images x_1 and x_2, each image is processed using the feature extraction network to generate feature vectors, f_1 and f_2. Additionally, we refer to their subject identification labels as y_1 and y_2 respectively. Hence, we can write the Siamese networks training objective as a function of the feature vectors as follows:

$$Verification_{loss} = \begin{cases} \frac{1}{2}\|f_1 - f_2\|^2 & y_1 = y_2 \\ \frac{1}{2}[max(m - \|f_1 - f_2\|, 0)]^2 & y_1 \neq y_2 \end{cases} \tag{5.3}$$

where $\|f_1 - f_2\|^2$ is the Euclidean distance between the feature vectors. When the features pair are from the same person i.e., $y_1 = y_2$, the objective encourages the features f_1 and f_2 to be close. In contrast, for the features pair from different persons i.e.,$y_1 \neq y_2$ the objective encourages the features to be separated by a margin m.

The second signal is identification loss function which is achieved by feeding the features to softmax layer. This layer classifies each image into one of K different identities (e.g., $K = 100$) by giving the probability distribution over the K classes. So given a pair of features (f_1 and f_2) with their associated subject identities (y_1, y_2), we can derive the probability distribution as follows:

$$P_1 = P(y_1 = i|f_1) = \frac{\exp^{W_i f_1}}{\sum_k \exp^{W_k F_1}} \tag{5.4}$$

$$P_2 = P(y_2 = j|f_2) = \frac{\exp^{W_j f_2}}{\sum_k \exp^{W_k F_2}} \tag{5.5}$$

where W_i and W_j refer to the ith and jth column of softmax weight matrix, respectively.

The network is then trained to minimize the cross-entropy loss (identification-loss) as defined below:

$$Ident_1 = -\sum_{i=1}^{K} y_1^i \log P_1^i \tag{5.6}$$

$$Ident_2 = -\sum_{j=1}^{K} y_2^i \log P_2^j \tag{5.7}$$

where $y_1, y_2 = 0$ are for all i ,j except for the target class. In order to generate an effective verification system, we need to enhance the system with the identification signal that has very rich inter-class variation. We noticed that the early activating of this signal will improve the separation of the classes. Then after several epochs, this signal will vanish through controlled hyperparameter, while the verification signal will be activated gradually. Therefore, we can define the overall training loss function for a single pair of inputs, which jointly optimizes the verification cost and the identification cost as follows:

$$Total_{loss} = \gamma Verification_{loss} + (1-\gamma)(Ident_1 + Ident_2), \gamma : 0 \rightarrow 1 \tag{5.8}$$

$$\gamma = \frac{1}{\exp \frac{epoch_{num} - shift}{10}} \tag{5.9}$$

5.4 Experiments and Results

The Cifar100 dataset Krizhevsky et al. (2009) is a computer vision dataset established for object recognition problem. It has 100 classes, and each class has 600 color images, 500 images for training, and 100 images for testing. The size of the images is set to 32×32 pixels. Each image has two labels, a fine label that represents the class to which it belongs and a coarse label that represents the super class to which it belongs. Table 5.2 shows the list of classes included within each super class. However, in our experiments, we used the fine labels as a training label along with the images for the identification model.

All the images are resized to 64×64 pixels. A zero-centralization process is performed on our data to center the cloud of data around the origin, which involves subtracting the

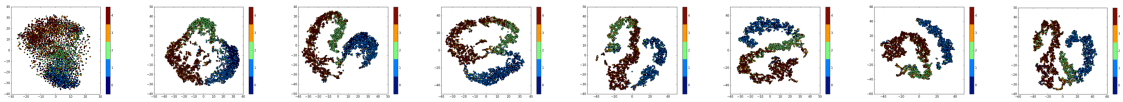
Table 5.2: Cifar-100 super classes.

Super class	Classes
aquatic mammals	beaver, dolphin, otter, seal, whale
fish	aquarium fish, flatfish, ray, shark, trout
flowers	orchids, poppies, roses, sunflowers, tulips
food containers	bottles, bowls, cans, cups, plates
fruit and vegetables	apples, mushrooms, oranges, pears, sweet peppers
household electrical devices	clock, computer keyboard, lamp, telephone, television
household furniture	bed, chair, couch, table, wardrobe
insects	bee, beetle, butterfly, caterpillar, cockroach
large carnivores	bear, leopard, lion, tiger, wolf
large man-made outdoor things	bridge, castle, house, road, skyscraper
large natural outdoor scenes	cloud, forest, mountain, plain, sea
large omnivores and herbivores	camel, cattle, chimpanzee, elephant, kangaroo
medium-sized mammals	fox, porcupine, possum, raccoon, skunk
non-insect invertebrates	crab, lobster, snail, spider, worm
people	baby, boy, girl, man, woman
reptiles	crocodile, dinosaur, lizard, snake, turtle
small mammals	hamster, mouse, rabbit, shrew, squirrel
trees	maple, oak, palm, pine, willow
vehicles 1	bicycle, bus, motorcycle, pickup truck, train
vehicles 2	lawn-mower, rocket, streetcar, tank, tractor

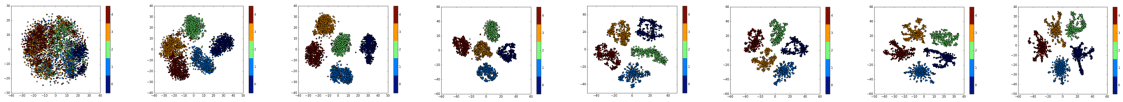
mean across all three color channels. Moreover, the data dimensions are normalized so that they are of approximately the same scale, and the minimum/maximum along each dimension is 0 and 1, respectively.

The network is trained using a batch size of 128. We use Adam optimizer Kingma and Ba (2014) as an adaptive learning rate optimization algorithm for training our deep neural networks. The baseline learning rate is set to 0.01 and decreased by a factor of 0.1 when the metric stops improving after ten epochs. To get the best computational performance, we use TensorFlow as the most popular and efficient machine learning tool for training our network. Moreover, we run our experiments on two NVIDIA 1080 Ti GPUs (Graphics Processing Unit) as underlying computing devices.

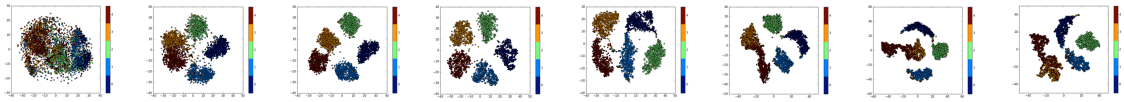
We use the t-distributed stochastic neighbor embedding (t-SNE), as shown in figure 5.2 to visualize only five classes from our dataset to explore further what occurs in the Euclidean Distance space. For the verification model shown in figure 5.2a, we noticed that the clustering does not constructed properly. Even on later epochs, the pattern is the same, except it will adopt different shapes. Therefore, we concluded that using the verification signal alone is not very effective in keeping the same identity features close, or keeping the different identities features far apart. In contrast, the learned features at identification space are shown to have large inter-class variation, as shown in figure 5.2b. This is attributed to the supervisory identification signal that tends to pull apart the features of different identities since they have to be classified into different classes. However, for this model, the Siamese model is trained for the identification task, not for the verification task. Figure 5.2c shows the feature embedding space in which the identification signal contributes along with the first four epoch, while the last five epochs were extracted from later epochs at which the verification signal was involved. We can see clearly that the combined model performs better than using only the verification model. Thus, satisfying both the verification loss and identification loss by jointly training is crucial for convergence.



(a) Verification Signal



(b) Identification Signal



(c) Both signals

Figure 5.2: Visualization of 2-dimensional features for only five classes of Cifar-100 dataset. (a) Illustrates the feature space when using only a verification signal. It has small intra-class variation; on the other hand, the inter-class variation is also small, which causes the different classes to be close to each other. (b) Illustrates the feature space when using only the identification signal. There is a large inter-class variation; on the other hand, the intra-class variation is not too small, which causes the identities belonging to the same class to be far apart from each other. (c) Illustrates the feature space when using both signals, the features have large inter-class variation and on the same time, the intra-class variation is small enough to keep the identities belong to same class close to each other.

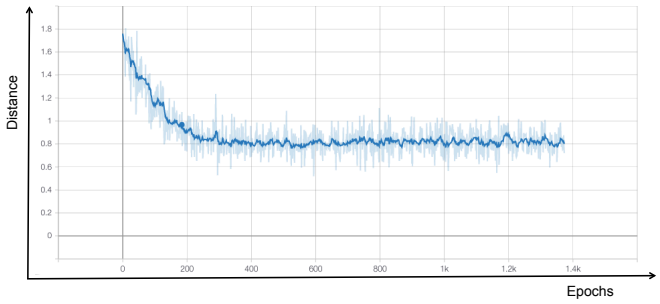


Figure 5.3: The distance between similar examples.

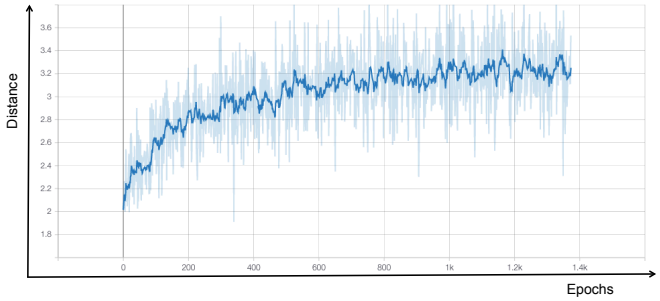


Figure 5.4: The distance between dissimilar examples.

In order to further investigate the inter/intra-class variation within our examples during network training, we track the changes of the similar distance and dissimilar distance while we are training the network. Specifically, we measure the distance between the pairs of similar inputs (the inputs that belong to the same class) and the distance between the pairs of dissimilar inputs (the inputs that belong to different classes). Then we average the distances and update this measure at each epoch. Figure 5.3 and Figure 5.4 show that we are successfully reducing the distance between similar inputs and increase the distance between dissimilar inputs.

Finally, we investigate the verification accuracy using three models. The first model is the identification model, where the Siamese networks are trained for the classification task, then we measured the verification accuracy depending on their class labels. The second model involves training our Siamese networks on the verification task, in which the similarity label used along with the images to determine if the two images are similar or dis-

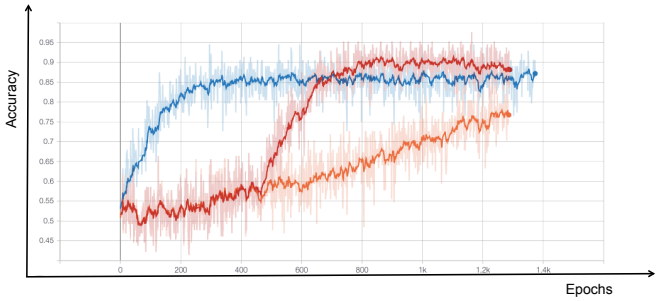


Figure 5.5: Training Accuracy.

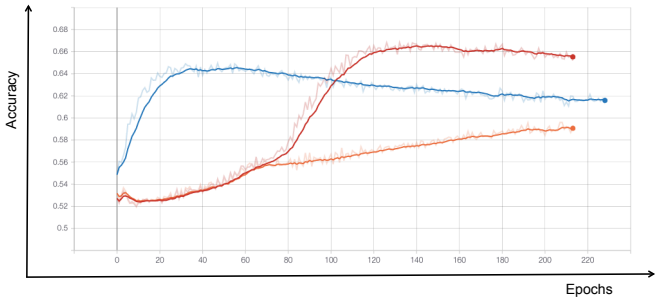


Figure 5.6: Validation Accuracy.

similar. In the third model, we combine the verification model and the identification model within the Siamese architecture. As we explained in Section 5.3.2 we integrate the identification signal first, then the verification model is integrating gradually. The results shown in Figures 5.5 and 5.6 are interesting as they show that joint identification-verification model surpasses the performance of both models alone.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

This Ph.D. dissertation presented an end-to-end deep Siamese networks for facial expression recognition (FER), valence and arousal estimation, and visual object matching. The facial expression Siamese model is aware of the local structure of the embedding space and gradually modulates the learning from the local adaptive verification space into the identification space. The verification model reduced the intra-class variation by minimizing the distance between the extracted features from the same identity using different strategies. In contrast, the identification model increased the inter-class variation by maximizing the distance between the features extracted from different identities.

We proved that applying the verification signal first and gradually integrating the identification signal into Siamese model leads to a better performance and aids in better generalization for identification task. The early activation of the verification signal improved the clustering of dataset classes. Consequently, after several epochs, the verification signal vanishes through controlled hyperparameter, while the identification signal is activated gradually.

We also showed how the mining strategy plays an important role to enhance the training efficiency and the system performance. For example, it showed that using the local mean or regularized feature in our proposed architecture to represent the whole class leads to a better performance than using the global mean or the original SML model. Therefore, using the local mean or regularized feature: compensated the dependency on the global mean, helped to maximize the distance between samples from the each class and the local mean, and thus assisted maximizing the inter-class variation. Moreover, our mining strategy showed that using only positive pairs or positive pairs alongside with all possible negative pairs performs less than our mining strategy of considering only the close classes (classes close to each other like Disgust and fear) alongside with positive pairs. Results were evaluated in three standard datasets for facial emotion recognition and surpass other methods in the metric learning area.

On the other hand, the empirical results of valence and arousal Siamese model demonstrated that transferring the learning from the classification space to the regression space enhanced the regression task since each expression occupies a representation within a specified range of valence-arousal affect. Further, for this model, we introduced several novel strategies for engaging the two spaces within the Siamese networks, ranging from different classification models to different methods for transfer the learning from one space to another. We concluded that using the FER classification model and clustering the embedding features into more number of classes (eight classes) results in better features representation than clustering the embedding features into four or two classes. However, there was no significant difference between the performances when using four- or two-class models. We also showed that loading weights of the eight-class classification model into Siamese networks is better than learning from scratch or loading weights of the regression model into Siamese networks. In contrast, loading the weights of four- or two-class models into Siamese networks performed less than loading the regression model into Siamese networks.

In our last Siamese model for image matching, we depend on the Siamese networks to learn directly from the pair of images, but we enhanced it with the identification signals. Accurately, we transfer the learning from the identification framework into the verification framework. Jointly learning both frameworks gave a better model performance since the identification framework helped to increase the inter-class variation in the verification framework.

6.2 Future Work

The preceding results fulfilled in applying the current methodology of Siamese networks will lead a further study in this direction. For example, in addition to facial expression recognition, valence/arousal estimation, and image matching tasks, it could be interesting to consider a person re-identification system based on deep convolutional features extracted from Siamese networks. A person re-identification system aims to retrieve or recapture a person of interest across multiple non-overlapping cameras or from the same camera in different occasions in an uncontrolled setting. These types of tasks should be evaluated on real-world unconstrained faces that have a full pose and illumination variation. IJB-A dataset is one of the suggestions to use, which captures a wide range of variation and offers challenges to face detection and face recognition systems. Additionally, it explores the subject-specific modeling in which it has been designed in a template-based manner. A template represents a collection of media (image and/or video frames) of an interesting subject captured in an uncontrolled setting. This strategy was inspired by many real-world high profile biometric scenarios. For example, the FBI's most wanted list has various facial images and video frames from subjects with several different viewpoints. Hence, there are several pooling and fusion ideas to generate one common vector representation, which can be broadly divided into three categories: (1) Feature pooling, (2) Similarity score pooling,

and (3) Image pooling. The template-based approaches also can be handled by building aggregation model like attention mechanisms into Siamese model. After extracting the features of every single media from the template, the aggregation model combine the best representation for each template. This technique will make the architecture learn from the most useful information that can produce a potential large magnitude of the gradient. Finally, one of the challenges that should be handled thoroughly; there will be no overlapping between the training and testing subjects to prevent any trial for subject-specific modeling. In this case, it is impossible to rely on any classification frameworks that have softmax layer to enhance the Siamese network's objectives. Softmax layer will learn to map the training facial images into different high dimensional space than that of the testing set due to the non-overlapping issue.

In other words, an effective person re-identification system on the IJB-A dataset requires first handling the extreme variation in this dataset. More precisely, a powerful discriminative model needed to be adopted to increase the inter-class variation and decrease the intra-class variation as the variation within the same class could overwhelm the differences between classes and make the face recognition more challenging. Second, it requires handling the set-to-set matching problem by generating one common feature representation for each set without increasing the computational and storage cost of template pair comparison. Finally, it requires capturing the best representation of the training set and finding a way to evaluate the trained networks on a testing set subjects that are non-overlapping with the training set subjects.

Bibliography

Challenges in representation learning: Facial expression recognition challenge. http://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge. 12, 27

A. Al-Hamadi, A. Saeed, R. Niese, S. Handrich, and H. Neumann. Emotional trace: Mapping of facial expression to valence-arousal space. *Current Journal of Applied Science and Technology*, pages 1–14, 2016. 44

A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky. Neural codes for image retrieval. In *European conference on computer vision*, pages 584–599. Springer, 2014. 64, 66

P. Barros, G. Parisi, and S. Wermter. A personalized affective memory model for improving emotion recognition. In *International Conference on Machine Learning*, pages 485–494, 2019. 45

S. Bermejo and J. Cabestany. Oriented principal component analysis for large margin classifiers. *Neural Networks*, 14(10):1447–1461, 2001. 50

B. Bhattarai, G. Sharma, and F. Jurie. Cp-mtml: Coupled projection multi-task metric learning for large scale face retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4226–4235, 2016. 7

Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2559–2566. Citeseer, 2010. 63

J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah. Signature verification using a" siamese" time delay neural network. In *Advances in Neural Information Processing Systems*, pages 737–744, 1994. 5

Q. Cao, Y. Ying, and P. Li. Similarity metric learning for face recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2408–2415, 2013. 2, 7, 33, 34

S. Cao and N. Snavely. Learning to match images in large-scale collections. In *European Conference on Computer Vision*, pages 259–270. Springer, 2012. 65

Y. Cao, C. Wang, L. Zhang, and L. Zhang. Edgel index for large-scale sketch-based image search. In *CVPR 2011*, pages 761–768. IEEE, 2011. 63

R. Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997. 46

V. Chandrasekhar, J. Lin, O. Morere, H. Goh, and A. Veillard. A practical guide to cnns and fisher vectors for image instance retrieval. *Signal Processing*, 128:426–439, 2016. 64, 66

W.-Y. Chang, S.-H. Hsu, and J.-H. Chien. Fatauva-net: An integrated deep learning framework for facial attribute recognition, action unit detection, and valence-arousal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 17–25, 2017. 42, 46

G. Chechik, U. Shalit, V. Sharma, and S. Bengio. An online algorithm for large scale image

similarity learning. In *Advances in Neural Information Processing Systems*, pages 306–314, 2009. 2, 7, 15

G. Chechik, V. Sharma, U. Shalit, and S. Bengio. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research*, 11(Mar):1109–1135, 2010. 2, 63

S. Chen, Q. Jin, J. Zhao, and S. Wang. Multimodal multi-task learning for dimensional and continuous emotion recognition. In *Proceedings of the 7th Annual Workshop on Audio/Visual Emotion Challenge*, pages 19–26, 2017. 44, 45

F. Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 38

S. Chopra, R. Hadsell, Y. LeCun, et al. Learning a similarity metric discriminatively, with application to face verification. In *CVPR (1)*, pages 539–546, 2005. 14

D.-A. Clevert, T. Unterthiner, and S. Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015. 69

Y. Cui, F. Zhou, Y. Lin, and S. Belongie. Fine-grained categorization and dataset bootstrapping using deep metric learning with humans in the loop. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1153–1162, 2016. 7

Z. Cui, W. Li, D. Xu, S. Shan, and X. Chen. Fusing robust face region descriptors via multiple metric learning for face recognition in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3554–3561, 2013. 2, 7

J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning.

In *Proceedings of the 24th international conference on Machine learning*, pages 209–216. ACM, 2007. 2, 7

X. Dong and J. Shen. Triplet loss in siamese network for object tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 459–474, 2018. 14

H. Drucker, C. J. Burges, L. Kaufman, A. J. Smola, and V. Vapnik. Support vector regression machines. In *Advances in neural information processing systems*, pages 155–161, 1997. 62

S. Du, Y. Tao, and A. M. Martinez. Compound facial expressions of emotion. *Proceedings of the National Academy of Sciences*, 111(15):E1454–E1462, 2014. iii, 12, 27, 38

Y. Duan, W. Zheng, X. Lin, J. Lu, and J. Zhou. Deep adversarial metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2780–2789, 2018. 7, 8

P. Ekman and W. V. Friesen. Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2):124, 1971. 40

R. Ekman. *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA, 1997. 40

H. Fan and H. Ling. Siamese cascaded region proposal networks for real-time visual tracking. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 7952–7961, 2019. 5

A. Frome, Y. Singer, and J. Malik. Image retrieval and classification using local distance functions. In *Advances in neural information processing systems*, pages 417–424, 2007. 63

R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *conference on computer vision and pattern recognition*, 2014. 44

A. Globerson and S. T. Roweis. Metric learning by collapsing classes. In *Advances in neural information processing systems*, pages 451–458, 2006. 2, 7

M. Glodek, S. Tschechne, G. Layher, M. Schels, T. Brosch, S. Scherer, M. Kächele, M. Schmidt, H. Neumann, G. Palm, et al. Multiple classifier systems for the classification of audio-visual emotional states. In *International Conference on Affective Computing and Intelligent Interaction*, pages 359–368. Springer, 2011. 43

J. Goldberger, G. E. Hinton, S. T. Roweis, and R. R. Salakhutdinov. Neighbourhood components analysis. In *Advances in neural information processing systems*, pages 513–520, 2005. 2, 7

I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D.-H. Lee, et al. Challenges in representation learning: A report on three machine learning contests. In *International Conference on Neural Information Processing*, pages 117–124. Springer, 2013. iii

A. Graves, N. Jaitly, and A.-r. Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *2013 IEEE workshop on automatic speech recognition and understanding*, pages 273–278. IEEE, 2013. 44

M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *International conference on Computer Vision*, 2009. 2, 7

Y. Guo, Y. Xia, J. Wang, H. Yu, and R.-C. Chen. Real-time facial affective computing on mobile devices. *Sensors*, 20(3):870, 2020. 45, 62

R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006. 14, 15, 64, 66

X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg. Matchnet: Unifying feature and metric learning for patch-based matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3279–3286, 2015. 65, 66

S. Handrich, L. Dinges, F. Saxen, A. Al-Hamadi, and S. Wachmuth. Simultaneous prediction of valence/arousal and emotion categories in real-time. In *2019 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pages 176–180. IEEE, 2019. 41, 46

B. Harwood, B. Kumar, G. Carneiro, I. Reid, T. Drummond, et al. Smart mining for deep metric learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2821–2829, 2017. 7, 14, 15

B. Hasani and M. H. Mahoor. Facial affect estimation in the wild using deep residual and convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 9–16, 2017. 44

B. Hasani, P. S. Negi, and M. H. Mahoor. Bounded residual gradient networks (breg-net) for facial affect computing. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–7. IEEE, 2019. 37, 38, 44, 45, 60, 62

W. Hayale, P. Negi, and M. Mahoor. Facial expression recognition using deep siamese neural networks with a supervised loss function. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*. 9, 44, 46, 47

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In

Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016a. 38

K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016b. 44, 61

L. He, D. Jiang, L. Yang, E. Pei, P. Wu, and H. Sahli. Multimodal affective dimension prediction using deep bidirectional long short-term memory recurrent neural networks. In *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge*, pages 73–80, 2015. 43

A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 38

J. Hu, J. Lu, and Y.-P. Tan. Discriminative deep metric learning for face verification in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014. 7, 8

J. Hu, J. Lu, and Y.-P. Tan. Deep transfer metric learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 325–333, 2015. 7

C. Huang, C. C. Loy, and X. Tang. Local similarity-aware deep feature embedding. In *Advances in neural information processing systems*, pages 1262–1270, 2016. 7, 8

G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017. 11, 28, 48, 56

A. Iscen, G. Tolias, Y. Avrithis, and O. Chum. Mining on manifolds: Metric learning without labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7642–7651, 2018. 7, 8, 14, 15

Y. Jang, H. Gunes, and I. Patras. Registration-free face-ssd: Single shot analysis of smiles, facial attributes, and affect in the wild. *Computer Vision and Image Understanding*, 182: 17–29, 2019. 47, 62

P. Khorrami, T. Le Paine, K. Brady, C. Dagli, and T. S. Huang. How deep neural networks can improve emotion recognition on video data. In *2016 IEEE international conference on image processing (ICIP)*, pages 619–623. IEEE, 2016. 41, 44, 45

B.-K. Kim, J. Roh, S.-Y. Dong, and S.-Y. Lee. Hierarchical committee of deep convolutional neural networks for robust facial expression recognition. *Journal on Multimodal User Interfaces*, 10(2):173–189, 2016. 38

J. C. Kim, H. Rao, and M. A. Clements. Investigating the use of formant based features for detection of affective dimensions in speech. In *International Conference on Affective Computing and Intelligent Interaction*, pages 369–377. Springer, 2011. 41

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 28, 57, 74

S. Kiran Yelamarthi, S. Krishna Reddy, A. Mishra, and A. Mittal. A zero-shot framework for sketch based image retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 300–317, 2018. 6

S. R. Klomp, D. W. van de Wouw, V. BV, et al. Ecdnet: Efficient siamese convolutional network for real-time small object change detection from ground vehicles. *Electronic Imaging*, 2019(7):458–1, 2019. 6

G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2, 2015. 6

D. Kollias and S. Zafeiriou. Exploiting multi-cnn features in cnn-rnn based dimensional emotion recognition on the omg in-the-wild dataset. *arXiv preprint arXiv:1910.01417*, 2019a. 41, 44, 45

D. Kollias and S. Zafeiriou. Expression, affect, action unit recognition: Aff-wild2, multi-task learning and arcface. *arXiv preprint arXiv:1910.04855*, 2019b. 41, 47

D. Kollias, M. A. Nicolaou, I. Kotsia, G. Zhao, and S. Zafeiriou. Recognition of affect in the wild using deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 26–33, 2017. 44, 45

D. Kollias, V. Sharmanska, and S. Zafeiriou. Face behavior\a la carte: Expressions, affect and action units in a single network. *arXiv preprint arXiv:1910.11111*, 2019a. 41, 42, 47

D. Kollias, P. Tzirakis, M. A. Nicolaou, A. Papaioannou, G. Zhao, B. Schuller, I. Kotsia, and S. Zafeiriou. Deep affect prediction in-the-wild: Aff-wild database and challenge, deep architectures, and beyond. *International Journal of Computer Vision*, 127(6-7): 907–929, 2019b. 44, 45

A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. iii, 72

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 2012. 44, 69

B. Kumar, G. Carneiro, I. Reid, et al. Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions. In *Proceedings*

of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5385–5394, 2016. 6, 7

B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 4282–4291, 2019. 5

J. Li, Y. Chen, S. Xiao, J. Zhao, S. Roy, J. Feng, S. Yan, and T. Sim. Estimation of affective level in the wild with multiple memory networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8, 2017. 41, 44, 45

Z. Li, S. Chang, F. Liang, T. S. Huang, L. Cao, and J. R. Smith. Learning locally-adaptive decision functions for person verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3610–3617, 2013. 7

A. Lindt, P. Barros, H. Siqueira, and S. Wermter. Facial expression editing with continuous emotion labels. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–8. IEEE, 2019. 41, 44, 45, 62

W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 212–220, 2017. 2, 7, 8

D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 65

J. Lu, X. Zhou, Y.-P. Tan, Y. Shang, and J. Zhou. Neighborhood repulsed metric learning for kinship verification. *IEEE transactions on pattern analysis and machine intelligence*, 36(2):331–345, 2013. 2, 7, 8

J. Lu, G. Wang, W. Deng, P. Moulin, and J. Zhou. Multi-manifold deep metric learning for image set classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1137–1145, 2015. 7, 8

X. Lu, W. Wang, C. Ma, J. Shen, L. Shao, and F. Porikli. See more, know more: Unsupervised video object segmentation with co-attention siamese networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3623–3632, 2019. 6

N. McLaughlin, J. M. del Rincon, and P. Miller. Recurrent convolutional network for video-based person re-identification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2, 9, 25, 36

M. Mehu and K. R. Scherer. Emotion categories and dimensions in the facial communication of affect: An integrated approach. *Emotion*, 15(6):798, 2015. 46

I. Melekhov, J. Kannala, and E. Rahtu. Siamese network features for image matching. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 378–383. IEEE, 2016. 64, 66

H. Meng and N. Bianchi-Berthouze. Naturalistic affective expression classification by a multi-stage approach based on hidden markov models. In *International Conference on Affective Computing and Intelligent Interaction*, pages 378–387. Springer, 2011. 43

H. Meng, D. Huang, H. Wang, H. Yang, M. Ai-Shuraifi, and Y. Wang. Depression recognition based on dynamic facial and vocal expression features using partial least square regression. In *Proceedings of the 3rd ACM international workshop on Audio/visual emotion challenge*, pages 21–30, 2013. 43

A. Mollahosseini, D. Chan, and M. H. Mahoor. Going deeper in facial expression recognition using deep neural networks. In *2016 IEEE Winter conference on Applications of Computer Vision (WACV)*, pages 1–10. 38

A. Mollahosseini, B. Hasani, and M. H. Mahoor. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing*, 2017. iii, 12, 13, 27, 37, 44, 49, 56

Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh. No fuss distance metric learning using proxies. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 360–368, 2017. 7, 8

J. Mueller and A. Thyagarajan. Siamese recurrent architectures for learning sentence similarity. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016. 6

P. Neculoiu, M. Versteegh, and M. Rotaru. Learning text similarity with siamese recurrent networks. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 148–157, 2016. 6

H. V. Nguyen and L. Bai. Cosine similarity metric learning for face verification. In *Asian conference on computer vision*, pages 709–720. Springer, 2010. 2, 7

M. A. Nicolaou, H. Gunes, and M. Pantic. Continuous prediction of spontaneous affect from multiple cues and modalities in valence-arousal space. *IEEE Transactions on Affective Computing*, 2(2):92–105, 2011. 45, 50

J. Nicolle, V. Rapp, K. Bailly, L. Prevost, and M. Chetouani. Robust continuous prediction of human emotions using multiscale dynamic cues. In *Proceedings of the 14th ACM international conference on Multimodal interaction*, pages 501–508, 2012. 43

H. Oh Song, Y. Xiang, S. Jegelka, and S. Savarese. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 8, 9, 14

H. Oh Song, S. Jegelka, V. Rathod, and K. Murphy. Deep metric learning via facility location. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5382–5390, 2017. 8

S. Paisitkriangkrai, C. Shen, and A. Van Den Hengel. Learning to rank in person re-identification with metric ensembles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1846–1855, 2015. 8

S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009. 46

G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. Murphy. Towards accurate multi-person pose estimation in the wild. In *Conference on Computer Vision and Pattern Recognition*, 2017. 44

O. M. Parkhi, A. Vedaldi, A. Zisserman, et al. Deep face recognition. In *British Machine Vision Conference*, 2015. 8

A. M. Qamar and E. Gaussier. Online and batch learning of generalized cosine similarities. In *2009 Ninth IEEE International Conference on Data Mining*, pages 926–931. IEEE, 2009. 2, 7, 26, 36

A. M. Qamar, E. Gaussier, J.-P. Chevallet, and J. H. Lim. Similarity learning for nearest neighbor classification. In *2008 Eighth IEEE International Conference on Data Mining*, pages 983–988. IEEE, 2008. 2, 7

Y. Qi, Y.-Z. Song, H. Zhang, and J. Liu. Sketch-based image retrieval via siamese convolutional neural network. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 2460–2464. IEEE, 2016. 6

G. A. Ramirez, T. Baltrušaitis, and L.-P. Morency. Modeling latent discriminative dynamic of multi-dimensional affective signals. In *International Conference on Affective Computing and Intelligent Interaction*, pages 396–406. Springer, 2011. 43

J. Redmon and A. Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017. 46

J. A. Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161, 1980. 40, 41, 42

F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015. 8, 14

S. Shalev-Shwartz, Y. Singer, and A. Y. Ng. Online and batch learning of pseudo-metrics. In *Proceedings of the twenty-first international conference on Machine learning*, page 94. ACM, 2004. 2, 7

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 38

H. Siqueira. An adaptive neural approach based on ensemble and multitask learning for affect recognition. In *Proceedings of the International PhD Conference on Safe and Social Robotics, Madrid, Spain*, pages 29–30, 2018. 41, 46

H. Siqueira, S. Magg, and S. Wermter. Efficient facial feature learning with wide ensemble-based convolutional neural networks. *arXiv preprint arXiv:2001.06338*, 2020. 45

J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *null*, page 1470. IEEE, 2003. 65

K. Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in Neural Information Processing Systems*, 2016. 8, 14

H. O. Song, S. Jegelka, V. Rathod, and K. Murphy. Deep metric learning via facility location. In *Computer Vision and Pattern Recognition*, 2017. 9, 14

Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *Advances in neural information processing systems*, 2014. 2, 9, 15, 65

F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018. 6

C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 38

C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017. 38

Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014. 6, 8

Y. Tang. Deep learning using linear support vector machines. *arXiv preprint arXiv:1306.0239*, 2013. 38

G. W. Taylor, I. Spiro, C. Bregler, and R. Fergus. Learning invariance through imitation. In *CVPR 2011*, pages 2729–2736. IEEE, 2011. 63, 64, 66

Y. Tian, B. Fan, and F. Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 661–669, 2017. 65, 66

J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler. Efficient object localization using convolutional networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 648–656, 2015. 6

E. Ustinova and V. Lempitsky. Learning deep embeddings with histogram loss. In *Advances in Neural Information Processing Systems*, pages 4170–4178, 2016. 8

M. F. Valstar, M. Mehu, B. Jiang, M. Pantic, and K. Scherer. Meta-analysis of the first facial expression recognition challenge. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(4):966–979, 2012. 1

G. K. Verma and U. S. Tiwary. Affect representation and recognition in 3d continuous valence–arousal–dominance space. *Multimedia Tools and Applications*, 76(2):2159–2183, 2017. 41

V. Vielzeuf, C. Kervadec, S. Pateux, and F. Jurie. The many variations of emotion. In *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–7. IEEE, 2019. 45

J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1386–1393, 2014a. 8, 64, 66

J. Wang, F. Zhou, S. Wen, X. Liu, and Y. Lin. Deep metric learning with angular loss. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2593–2601, 2017. 8

X.-Y. Wang, B.-B. Zhang, and H.-Y. Yang. Content-based image retrieval by integrating color and texture features. *Multimedia tools and applications*, 68(3):545–569, 2014b. 63

K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in neural information processing systems*, pages 1473–1480, 2006. 2, 7

Y. Wen, Z. Li, and Y. Qiao. Latent factor guided convolutional neural networks for age-invariant face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016a. 9

Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, 2016b. 8, 9

C. Wengert, M. Douze, and H. Jégou. Bag-of-colors for improved image search. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1437–1440. ACM, 2011. 63

P. Wu, S. C. Hoi, H. Xia, P. Zhao, D. Wang, and C. Miao. Online multimodal deep similarity learning with application to image retrieval. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 153–162. ACM, 2013. 64, 66

X. Wu, A. Kimura, S. Uchida, and K. Kashino. Prewarping siamese network: Learning local representations for online signature verification. In *ICASSP 2019-2019 IEEE In-*

ternational Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 2467–2471. 6

W. Xiaohua, P. Muzi, P. Lijuan, H. Min, J. Chunhua, and R. Fuji. Two-level attention with two-stage multi-task learning for facial emotion recognition. *Journal of Visual Communication and Image Representation*, 62:217–225, 2019. 46

W. Xie, L. Shen, and J. Duan. Adaptive weighting of handcrafted feature losses for facial expression recognition. *IEEE Transactions on Cybernetics*, 2019. 38

E. P. Xing, M. I. Jordan, S. J. Russell, and A. Y. Ng. Distance metric learning with application to clustering with side-information. In *Advances in neural information processing systems*, pages 521–528, 2003. 2, 7

J. Yang, P. Ren, D. Zhang, D. Chen, F. Wen, H. Li, and G. Hua. Neural aggregation network for video face recognition. In *CVPR*, 2017. 6

Z. Yu and C. Zhang. Image based static facial expression recognition with multiple deep network learning. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 435–442. ACM, 2015. 38

S. Zagoruyko and N. Komodakis. Learning to compare image patches via convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4353–4361, 2015. 64, 66

L. Zhang, S. Peng, and S. Winkler. Persemon: A deep network for joint analysis of apparent personality, emotion and their relationship. *IEEE Transactions on Affective Computing*, 2019. 47

X. Zhang, M. H. Mahoor, and R. D. Nielsen. On multi-task learning for facial action unit

detection. In *2013 28th International Conference on Image and Vision Computing New Zealand (IVCNZ 2013)*, pages 202–207. IEEE, 2013. 46

Y. Zhang and Q. Yang. A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*, 2017. 46

Z. Zhang and H. Peng. Deeper and wider siamese networks for real-time visual tracking. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 4591–4600, 2019. 5

M. Zheng, S. Karanam, Z. Wu, and R. J. Radke. Re-identification with consistent attentive siamese networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 5735–5744, 2019. 6

W.-S. Zheng, S. Gong, and T. Xiang. Person re-identification by probabilistic relative distance comparison. In *CVPR 2011*, pages 649–656. IEEE, 2011. 2, 7

S. Zhou, Y. Liang, J. Wan, and S. Z. Li. Facial expression recognition based on multi-scale cnns. In *Chinese Conference on Biometric Recognition*, pages 503–510. Springer, 2016. 38