A Study of Signature Authentication using Convolutional Neural Network and Transfer Learning Approach

Supervised by Dr. Yun Sing Koh

Jiachen Lyu
UID:jlu851

Henri Christiaan
UID:Hchr147

Colby Carillo
UID:ccar788

Bokai Chen
UID:bche392

***Abstract*—Abstract— In this study, we tested the feasibility of implementing transfer learning techniques by transferring a pre-trained Convolutional Neural Network (trained on Imagenet), onto the signature verification domain. We then compared the accuracy of this transfer learning approach to the accuracy of conventional CNN trained on the same signature dataset. Based on limited training and testing, it was found that certain implementations of transfer learning were able to outperform CNN in accuracy. However, to confirm this finding, further testing is required.**

I. Introduction

A. Introduction

Written signatures have many attributes of good identification biometrics, such as universality, uniqueness, permanence, and collectability [1]. With the ever growing technology in communication and security, signatures are still widely used as a means of identity verification across a variety of applications, such as contracts, banking, and driver's license. It has become an integrated part of most societies and is considered a requirement for all individuals within society upon reaching legal adulthood. Handwritten signatures verify the identity of the author by providing a signed representation of their name. To this day, despite the development of new identification biometrics, such as facial and iris authentication technologies, the written signature remains the most socially and legally accepted means for identification [2]. In our research, we aim to increase the accuracy of signature verification by transferring an architecture trained from ImageNet data to a few different datasets, ranging in quantity and language, to learn classification of genuine and forgery signatures via our selected Convolutional Neural Network.

B. Background knowledge

1) signature authentication task: Machine learning task strategies of signature verification could be mainly categorised into writer-independent learning (general learning) and writer-dependent learning (specialised learning) [3]. In the writer-dependent scenario, an individual classifier is trained independently for each writer, while in the case of writer-dependent, only one classifier is trained over all authors samples [3].

2) transfer learning: Transfer learning typically trains a neural network on a base dataset and task. This neural network is then repurposed to learn features from the original domain, then transfer them to a new task to be trained on a new task and dataset. This process is usually more successful if the training task and the new task share common features, and is less successful if the feature is specific to the training task [4]. There are a variety of transfer learning techniques that can be applied within and outside of machine learning. These techniques can be broken into three groups Homogeneous transfer learning, Heterogeneous transfer learning, and Negative transfer [5]. Homogeneous transfer learning is where the input feature space of the source domain is equal to that of the task domain, while heterogeneous is when these feature spaces are different [5].

3) convolutional neural network: Both our baseline model and transfer-learning models were trained with a convolutional neural network. LeCuCNN advanced in the following fields:

- Feature extraction: CNN extracts local features by taking synaptic inputs from a local 'receptive field' in the previous layer for each neuron.
- Feature mapping: Each convolutional layer has several feature maps, where individual neurons share

machine learning research on signature verification						
author	year	method	ERR	AER	advantage	limitations
Okawa	2018	fisher vector with fusion strategy	5.47%		1. low error rate while keep background(noise) features 2.flexibility that enables application to signatures with defective strokes	generalization ability has not been tested
Ooi et al	2016	DRT+PCA+PNN	9.87%		fast speed and high accuracy	not generalized solution
Bertolini et al	2010	ensemble		8.16%	save cost of retraining when adding new writer data	rely a lot on preprocessing in classifying feature parameters
research uses CNN & transfer learning						
author	year	method	ERR	AER	advantage	limitations
L. Hafemann, et al	2017	deep CNN+ Tranfser learning (Stochastic Gradient Descent with Nesterov Momentum)	1.72%		This method reduce the dependency for the classifier to rely skilled forgeries for training and significantly improve the accuracy of the classifiers.	suitable for the signature system with imbalanced writer-dependant sample between genuine and forgery
Soleimani,Araabi & Fouladi	2016	distance metric based CNN+transfer learning	12.80%		utilise similarity and dissimilarity between genuine and forgery signature to improve performance; the solution is more general about databse	the performance can be improved

Fig. 1. summary table of related work

the same set of synaptic weights in the same plane. Weight sharing allows for shift invariance and a dramatic decrease in the size of the free parameter set.

- Subsampling: Each convolutional layer has a sequential layer, which performs "local averaging and subsampling" to reduce the feature map [6].

Thus, a convolutional neural network is a multi-layer neural network frequently used to recognise images (2D shapes) with a high degree of shift invariance capability [7]. Furthermore, the free parameter size of CNN is reduced by means of stochastic backpropagation, which lowers the cost of learning and thus improves the generalisation ability of the learning. Convolutional networks also allow parallel implementation by weight sharing [7].

C. Hypothesis

Our hypothesis is that transfer learning techniques will improve the accuracy of signature authentication over a limited set of training signature data, and a tight time constraint. To test this hypothesis, the classifier developed through transfer learning will be compared against the traditional convolutional neural network trained on the training signature dataset, to identify forged signatures among genuine signatures. To optimise the transfer learning model, we selectively froze and unfroze different layers of the transferred model, while training the unfrozen layer on the signature data using backpropagation. This optimised transfer learning model is then compared to a CNN model, which is trained purely on the signature data. The accuracy of our classifiers are measured through Average Error Rate (AER), which is an approximation to Equal Error Rate (EER), which has been used consistently in Signature Authentication literature.

D. Motivation

In our research, we aim to improve the accuracy of signature verification by transferring an architecture trained from ImageNet data to a few different datasets, ranging in quantity and language, to learn classification of genuine and forgery signatures via our selected Convolutional Neural Network.

If our research is proved successful, many applications with a small number of signature data can use transfer learning classifiers to authenticate these signatures easily, which would not be otherwise possible due to the limited signature sample and training time.

In this report, section II will summarise exisitng work related to our project,section III will explain our methodology in details, section IV will discuss about the results and finally sectionV will conclude our project.

II. Related Work

The table above summarizes the related work and detailed description will expand out in this section.

A. Related work with tradictional machine learning techniques

Ooi et al. [8] proposed a framework for combining discrete Radon transform (DRT), principal component analysis(PCA), and probabilistic neural network(PNN) to identify signature forgery with image data. They used median filtering to remove the background noise, then binarised the image data. They created the data sample by collecting both 1000 genuine data, 500 casual forgeries and 500 skilled forgeries from 100 writers. The results show that the framework performs forgery detection task with higher FAR and lower FRR rate with the test sample from the same database. However, it doesn't show any significant improvement with test samples from a different database.

Okawa [9] proposed an approach using Fisher vector. This approach extracts KAZE features from both foreground and background signatures, with a fusion strategy to improve discriminative power. The result illustrated a significant improvement (1% significance level) of equal error compared with VLAD method and VLAD method with fusion strategy. They also found that the approach works well with background signature images, whose features are normally removed as noise in the preprocessing stages, for other machine learning algorithms. It was also discovered that FV has a more precise spatial distribution of the writing pattern for each author.

Bertolini et al. [10] applied ensemble classifiers on a signature data set to improve the reliability of a signature verification system. They trained ensemble classifiers on a different database of signatures to the test signature database. One dataset contains only genuine and random forgery signatures; the other contains only genuine signatures and all three types of forgeries respectively. The ensemble learning task involves learning four features, namely density, slant, distribution and curvature. The result of the classifier indicates that ensemble learning allows detections on new signature data set, without retraining classifiers by means of adjusting ensembles involved in the prediction model.

B. Related work with transfer learning approach

L. Hafemann et al. [2] selected signature features by first applying a CNN to writer-independent signatures using skilled forged signatures, which included both the identification of the users as well as determining if the signature is genuine. This step aims to establish a hypothesis space of signature features that can be used to distinguish between genuine and forged signatures. The selected features are then retrained (transfer learning) on a new set of writer-dependent signatures by unfreezing the fully-connected layer. In this step, only random forgeries (no skilled forgeries) were used. This method was able to achieve the state of the art signature authentication accuracy for single classifiers, and also reduced the dependency for the classifiers to rely on skilled forgeries for training. It was also noted that batch normalisation has proven to improve the accuracy of the classifiers significantly.

Soleimani et al. [11] looked at a solution to signature verification using a Deep Multi-task Metric Learning applied. Within all of their experiments, the DDML method proposed had a low Equal Error Rate (EER). It was found that the shared layer helps the network to learn a representation, that discovers underlying shared factors among signatures of different individuals, while distinct layers try to learn writer specific factors.

TABLE I
SUMMARY OF CNN TRAINING SETTINGS

Architecture	VGG-16	**max epoch size**	90
Learning rate	0.000005	**steps per epoch**	200
Loss function	Triplet	**Initial weights**	random small
Optimizer	Adam	**validation size**	3 X 3

III. METHODOLOGY

A. Datasets

[12]This project trains the model with UTSig datasets. UTSig has 115 classes containing: 27 genuine signatures; 3 opposite-hand signed samples and 42 simple forgeries. Each class belongs to one specific authentic person. UTSig has 8280 images collected from undergraduate and graduate students of the University of Tehran and Sharif University of Technology. Signatures were scanned with 600 dpi resolution and stored as 8-bit Tiff files. The signatures were written in Arabic.

Fig. 2. Sample signatures from UTSig datasets.Top rows are genuine signatures while the bottom row are forged [12].

B. Baseline model

Table I summarizes the settings for our baseline model. We will explain the choice for architecture, *1) VGG-16 architecture:* VGG-16 is a 16-layer CNN, that originates from the Visual Geometric Group of the University of Oxford. VGG is the first CNN which uses a 3x3 convoluted filters as oppose to bigger filters, such as the ones used in other CNN architectures before it. VGG seems to occupy the sweet spot between complexity and accuracy for what we are looking for in our research project. It is not a very deep CNN; however, it still has good ImageNet Validation accuracy [13] [14]

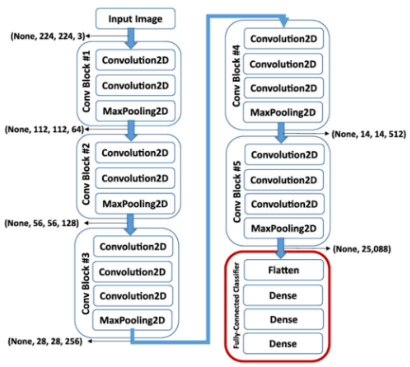


Fig. 3. Architecture diagram of VGG-16 [15]

2) tripliet Loss:

$$L = \sum_{i}^{N} \left[\left\| f(x_i^a) - f(x_i^p) \right\|_2^2 - \left\| f(x_i^a) - f(x_i^n) \right\|_2^2 + a \right]_+ \quad (1)$$

Equation (1) demonstrates the calculation formula for the triplet loss. Triplet Loss is a loss function where the distance between the same identity is minimised and the distance between different identities are maximised [16]. We adapted triplet loss function proposed by [17] for offline signature verification problem because it has been verified in [16] and we believe the task of offline signature verification by authors is similar to the task of face verification by identities while both tasks apply CNN structure for training.

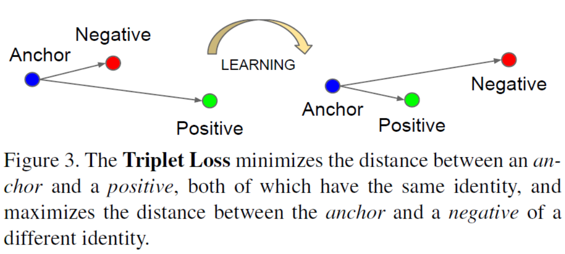


Fig. 4. Triplet Loss [16]

3) 3 X 3 validation: To ensure our classifiers do not overfit the training data, cross-validation is used in the training of our classifiers. Due to the limited computational resource and time constraint, we used a 3 X 3 fold cross-validation approach instead of the conventional 10 X 10 fold cross-validation approach.

The data was first randomised, then split into 3 mutually exclusive sections. Out of the three training data sections, one section is selected as the validation set, while the remaining two sections were employed as the training set. A classifier was fully trained on the training set, while the accuracy of the classifier was estimated on the validation set. Two more classifiers were trained by selecting different data sections for the validation set. After all three sections have been selected as the validation set, the data is then randomised, and the training process repeats. This randomisation process was carried out three times. As a result, the entire training process occurs nine times for each model.

After all nine classifiers are trained, the model with the lowest validation error is selected as the final classifier for the model to be tested on the test signature dataset. The result is measured and recorded. The three randomisation seeds used in the training process were 30, 123, and 234.

4) early stop strategy: To obtain more test results in the limited time frame of this project, an early stopping strategy was adopted. During training iteration of the classifier, the validation loss of the classifier was measured on the validation set. After each iteration of the training of the classifier, the validation loss is expected to decrease compared to the previous iteration. If the validation loss does not improve for more than a certain number of iterations, the training will stop.

The threshold for early stopping aws set to 50 iterations without an improvement of the validation loss. 50 is selected because it reduced the time required for training while not allowing the training to stop prematurely.

C. Transfer learning model

Transfer learning models applied all the same settings with the baseline model except for the initial weights. Transfer learning models used or start edwith the pretrained weights.

1) source task domain: Our pretrained weights source from the [18], detecting objects in 10 million labelled images and assigning them to 1000 object categories.

2) target task domain: The weights pretrained for Imagenet 2012 competition task were applied to the signature authentication task. Forgery signatures and

TABLE II
SUMMARY OF TRAINING STATISTICS

	Baseline	Transfer-15	Transfer-11	Transfer-7	Transfer-4	Transfer-0
training time (per epoch)	22s	13s	17s	19s	21s	22s
average stopping time (epochs)	83.7	88.7	78.9	76.9	73.7	78.3
average AER	36.41%	32.06%	33.51%	31.87%	36.65%	34.03%

genuine signatures for the same identity de facto share couples of common features, which is not the case for the object classification. We still expect an improvement in learning performance with the help of pretraining weights.

3) layers to transfer: There are 13 convolutional layers and 3 dense layers in VGG-16 architecture. Firstly, we only released the last dense layer for training and fixed the weights in the all remaining 15 layers. The model with 15 layers fixed will be denoted as ”transfer-15” later in this report.

[4] discovered that initialising with transferred weights at any layer will provide a general improvement in performance despite the dissimilarity between the learning objective and the source task after fine-tuning. Therefore, we adopted the fine-tuning strategy by unfreezing the convolutional training blocks, as displayed in FIg. 3, in back-propagation for retraining. The models with unfrozen retraining layers will be denoted as ”transfer-11”, ”transfer-7”,”transfer-4” and ”transfer-0” models. Transfer-0 distinguishes itself with the baseline model in initial weights. Baseline model training with random weights while transfer-0 model initiates with transferred weights. We expect all the transfer learning model to learn better in signature features than the baseline model.

D. Evaluation matrix

We plan to evaluate how successful our transfer is, by comparing it to a baseline model trained directly on the test datasets. Accuracy will be the primary measure of our classifiers. Below is the matrix that we are planning to use:

- FAR - Failed Acceptance Rate
- FRR - False Rejection Rate
- EER - Equal Error Rate (which is the error when FAR = FRR)
- AER - Average Error Rate

In signature Authentication literature, Equal Error Rate (EER) has been used consistently used to evaluate the accuracy of classifiers. EER is the error rate of the classifier when the False Rejection Ratio (FRR) is equal to its False Acceptance Ratio (FAR). However, to obtain EER, an extensive amount of testing is require. Due to the limited time frame and resource of this project, ERR cannot be obtained. Instead, Average Error Rate (AER) insert formula is as an approximation of ERR. AER is calculated by taking the average of FRR and FAR, and is sometimes used in literature as approximations for ERRs.

E. sensitivity test on learning rate

The learning rate used in the cross-validation is 0.000005. To understand the impact of learning rate on the training of the classifiers, simple tests (without validation tests) of learning 0.000001 and 0.00001 were also carried out.

IV. Experimental Results

We trained the baseline model and transfer-learning models with 90 epochs per training. Table II summarises the training details for each model.” training time per epoch” is the time cost per training epoch. Training time per epoch for ”transfer-15” model is significantly reduced compared with other models Since transfer-15 model fixed all the weights trained in the convolutional neural training layers and just reassigned the weights. ”average stopping time (epochs)” illustrates when the models decided to stop learning based on the validation loss history with patience of 50 epochs on average. There is a descending trend in epochs trained when we unfroze the convolutional training layer blocks backpropagation. It might contribute to the trend that the models were able to adjust weights to fit the training objective. Such behaviour might increase the efficiency in learning. ”average AER” takes the average of AERs obtained from our the three-times-three validation test results. We will discuss the learning performance in accuracy in IV-A.

A. Evaluation results

Fig. 5 demonstrates the learning results in error rates. The green, blue and yellow piles present the mean false acceptance error, the average false rejection rate and the average error rate (defined in III-D) respectively for

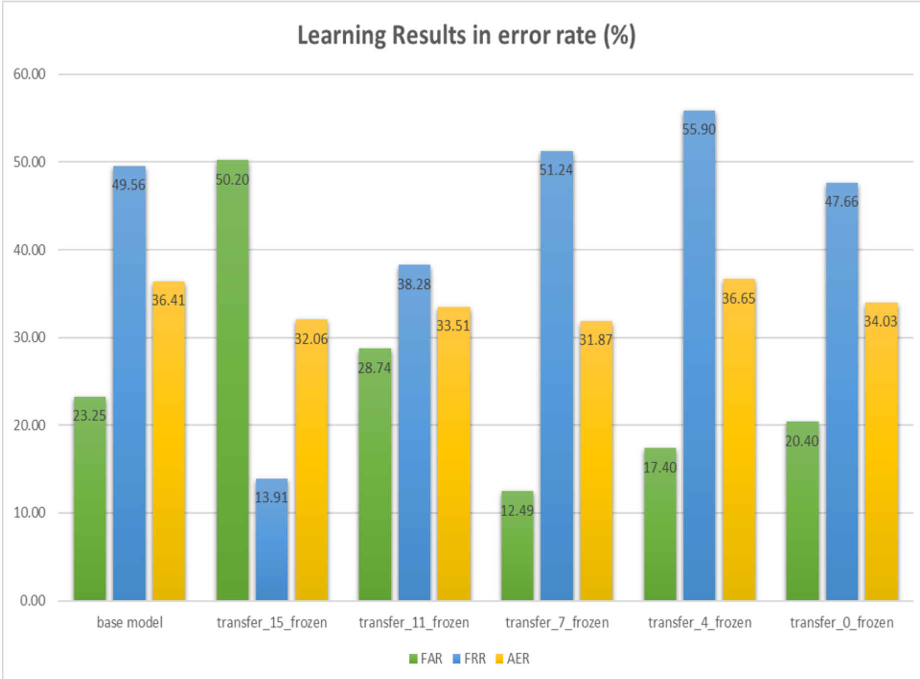


Fig. 5. Learning results in error rates

TABLE III
EXPERIMENTAL RESUTLS SUMMARY

	AER(%)	FAR(%)	FRR(%)
Best	transfer-7(31.9)	transfer-7(12.5)	transfer-15(13.9)
Average	transfer-11(33.5)	baseline(23.25)	baseline(49.6)
Worst	transfer-4(36.7)	transfer-15(50.2)	transfer-4(55.9)

each training models in the scale of percentages. Also, Table III summarizes the results with a description of which model achieves what level with our evaluation metric.

Compared with the baseline model, transfer learning improved AER in 4% when we transferred all the weights and fixed all the training layers. The FAR and FRR result patterns for the baseline model and the transfer-15 model are almost in mirror symmetry. The significantly high FAR in transfer-15 might be the consequence of freezing all the training layers as the neuron weights are not tailored to the signature learning.

We applied a fine-tuning strategy and expected to see significant improvement with it as proposed by [4]. Fig. 5 shows that transfer learning with our signature authentication task reaches the optimal accuracy, 31.87%, when we unfroze the 4th and the 5th convolutional training layer blocks (refer to Fig. 3). The transfer-11 model (only unfroze the last convolutional training layer block) has the most balanced error rates with FAR and FRR. Besides, transfer-0 model improved accuracy by 2% with initialising transferred weights as expected.

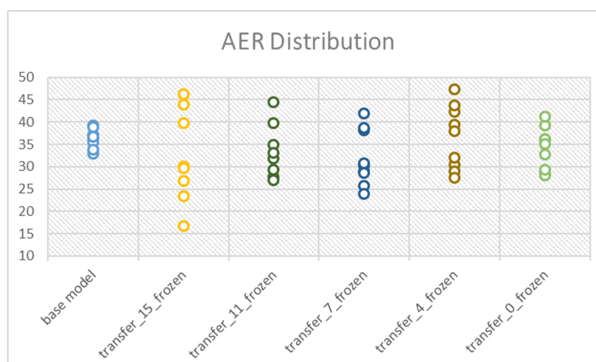


Fig. 6. Distribution of the validation results in AER

In summary, model transfer-7 is the best training model which benefits from its excellent performance in FAR. Moreover, model transfer-4 has the worst AER due to the high FRR. The baseline model, in fact, performs averagely among all the models. The results fail to fit our expectation of significant improvement in accuracy with the transfer learning approach contrast with our baseline model. Moreover, the transfer-4 model has worse accuracy than the baseline model. Besides, overfitting might have taken place in our training since FRR is approximately two to three times as FAR. Thus, we investigated our results further in IV-B.

B. Results investigation

From the results, it was found that the following factors have negative effects on the experiment results.

1) too few validation tests: Fig. 6 demonstrates the distribution of the validation observations in the metric of AER for each model. The distribution splits into two parts, the top and the bottom, for each training model with the transfer learning approach (transfer-15, transfer-11, transfer-7 and transfer-4). In fact, the bottom parts of the splits approximately match our expected pattern. The top parts are considered to be the outliers, happening by chance, and pulled up the error rates significantly due to the small size of validation test (9 in our project). To verify our conjecture, we need to conduct large scale of validation tests on that.

TABLE IV
OVERFITTING EXAMPLES STATISTICS

	Epochs trained	**AER**	**FRR**	**FAR**
baseline	90	36.4%	61.1%	11.6%
transfer-11	90	33.2%	61.1%	5.2%
transfer-7	89	30.3%	55.1%	5.1%
transfer-4	78	38.0%	71.8%	4.1%

2) overfitting problems: The FRR is significantly high in Fig. 5. It might be the result of overfitting. Thus, we reviewed the validation loss history. Fig. 7 demonstrates examples of overfittings. And Table IV summarises the statistics of the examples in the metrics of training epochs, AER,FRR and FAR. The baseline model example stops learning at 90th epoch out 90 epochs and the validation loss plot (Fig. 7(a)) Indicates an optimal stopping epoch around the 70th. Fig. 7(b), 7(c) and 7(d) all imply that they should stop around the 65th epoch while they stop at the 90th, 89th and 78th epochs respectively.

We run a trial test with the patience of 5 epochs before we started our experiment. It turned out to stop too early due to the fluctuation of validation loss. Therefore, we set the patience with 50 epochs. And now our experimental results imply waiting too long. And we fail to re-conduct the experiment with a smaller patience again due to the time constraints.

3) small datasets: Most of existing techniques discussed in section II report error rates below 20%. Therefore, we believe that our small dataset with around 8000 images contributes to the relatively insuffient learning performance.

C. Sensitivity test results

Table V summarises the sensitivity test results of the learning rate in the metric of AER. Our hypothesis that smaller learning rate will improve learning performance seems consistent across the table except for model Transfer-7 and model Transfer-0. We only run the sensitivity test once without any validation sets involved. Thus, we cannot draw our conclusion based on our simple trial.

V. Conclusion

A. Summary

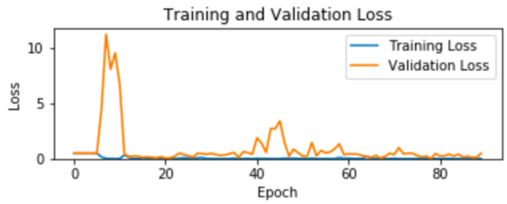
In our experiment, we transferred a CNN model trained on ImageNet to the domain of classifying the authenticity of offline signatures. We then compared the result of this transferred model to a baseline CNN model we have trained using only signature data. It was found that the transferred model performed similarly to the baseline model, with some implementations of the transferred model providing up to 4% higher accuracy compared to the baseline CNN model.

B. Limitation

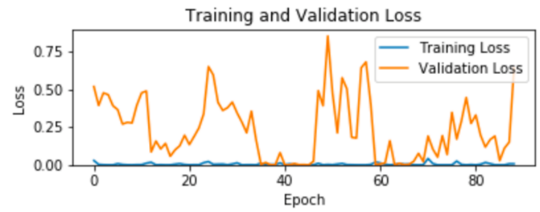
However, due to the size of the datasets we have as well as with the tight time constraints we have, we failed to fix the problems we have discussed in IV-B. Moreover, few validation sets involved in this project will weaken the strength of our conclusions.

C. Future Work

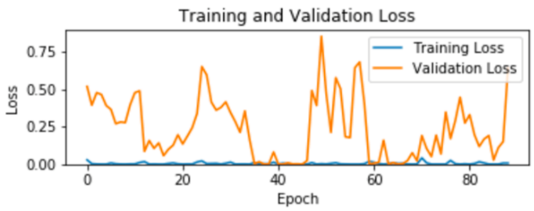
Much work is expected to be done in the future to investigate our hypothesis further. Firstly, we need to fix the issues we detected. Much work is required to be done in the future to investigate our hypothesis further. Firstly, more testing and validation will be carried out on the existing models. With more validation, the results will provide more confidence to the validity of the hypothesis of the project. Then, several parameters and structure of the algorithms need to be refined for optimum results. Due to the time constraint, many of these parameters are currently taken at default values from another project by two authors. Some of these potential changes include: **Fixing issues:**



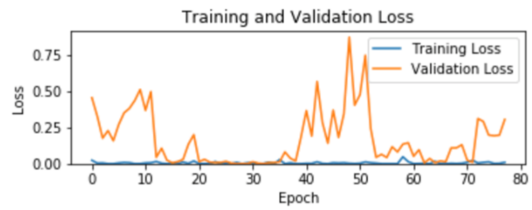
(a) baseline



(b) transfer-11



(c) transfer-7



(d) transfer-4

Fig. 7. examples of validation loss history

TABLE V
SENSTIVITY TEST OF LEARNING RATE (IN AER)

Baseline	**Baseline**	**Transfer-15**	**Transfer-11**	**Transfer-7**	**Transfer-4**	**Transfer-0**
0.000001	38.3%	25.6%	25.8%	35.8%	39.7%	47.3%
0.000005	39.3%	30.7%	29.3%	30.3%	42.3%	39.2%
0.00001	41.9%	32.06%	32.9%	40.6%	34.1%	41.6%

1) increase the times of validation: One of the straightforward ways to test our current hypothesis is to train the models more to get a stable distribution of error rates for both the baseline model and transfer-learning models. We believe the hypothesis can be verified to some degree when we have large size of validation test results.

2) change early stop setting: There is evidence of overfitting in the training of the classifiers. One source of overfitting comes from constraining the initial patience of epoch to 50. In the future, we will decrease the patience threshold to 20 or 15 for a more accurate comparison results.

3) increase the size of the datasets: Due to the time constraint, training and testing were only carried out on the UTsig dataset. This dataset, being Farsi signatures, may have had an effect on the results we observed. Therefore, by applying transfer learning to datasets of different size and variety, we may add value to our algorithms. Some publicly available datasets of interest include SIGCOMP 2009, SIGCOMP 2011, and CEDAR. Moreover, it may be possible that incorporating signatures in other languages such as English and Chinese to provide a more elaborate and an extensive test of our hypothesis.

Senstivity tests for learning performance:

4) learning rate: In this project, only one learning rate was implemented using the three by three cross-validation. The other two learning rates shown in the result section only ran on one model for every frozen block test. We would like to exhaustively test the learning rates to find the optimal learning rate for each model.

5) prediction threshold: The threshold determinant is 32 in this project, merely following the source code. We did a simple test on it, i.e. change 32 to 64. And the prediction accuracy was dramatically increased. We failed to conduct a formal sensitivity test on it due to the time constraints. Most of the literature claims the threshold as a random number. Therefore, we wouldd like to investigate more whether the surprising improvement just happened by chance or it does have influence with learning performance.

6) number of epoches: The number of epochs will be adjusted to observe its effect on the accuracy of the classifiers.

7) training architectures: In this project, only the VGG-16 architecture was used for both transfer learning and CNN. Other suitable architectures may be more suitable, these include AlexNet, Inception, Reznet, and VGG-19. VGG-16 was chosen for this project primarily due to its relatively few layers and parameters, which is expected to have a shorter training time. In future works, different architectures will be tested. The accuracy of the trained classifiers will be compared.

References

[1] D.-Y. Yeung, H. Chang, Y. Xiong, S. George, R. Kashi, T. Matsumoto, and G. Rigoll, "Svc2004: First international signature verification competition," in *International conference on biometric authentication*. Springer, 2004, pp. 16–22.

[2] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, "Learning features for offline handwritten signature verification using deep convolutional neural networks," *Pattern Recognition*, vol. 70, pp. 163–176, 2017.

[3] S. N. Srihari, A. Xu, and M. K. Kalera, "Learning strategies and classification methods for off-line signature verification," in *Ninth International Workshop on Frontiers in Handwriting Recognition*. IEEE, 2004, pp. 161–166.

[4] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in neural information processing systems*, 2014, pp. 3320–3328.

[5] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, p. 9, 2016.

[6] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[7] S. S. Haykin, S. S. Haykin, S. S. Haykin, K. Elektroingenieur, and S. S. Haykin, *Neural networks and learning machines*. Pearson education Upper Saddle River, 2009, vol. 3.

[8] S. Y. Ooi, A. B. J. Teoh, Y. H. Pang, and B. Y. Hiew, "Image-based handwritten signature verification using hybrid methods of discrete radon transform, principal component analysis and probabilistic neural network," *Applied Soft Computing*, vol. 40, pp. 274–282, 2016.

[9] M. Okawa, "Synergy of foreground–background images for feature extraction: Offline signature verification using fisher vector with fused kaze features," *Pattern Recognition*, vol. 79, pp. 480–489, 2018.

[10] D. Bertolini, L. S. Oliveira, E. Justino, and R. Sabourin, "Reducing forgeries in writer-independent off-line signature verification through ensemble of classifiers," *Pattern Recognition*, vol. 43, no. 1, pp. 387–396, 2010.

[11] A. Soleimani, B. N. Araabi, and K. Fouladi, "Deep multitask metric learning for offline signature verification," *Pattern Recognition Letters*, vol. 80, pp. 84–90, 2016.

[12] A. Soleimani, K. Fouladi, and B. N. Araabi, "Utsig: A persian offline signature dataset," *IET Biometrics*, vol. 6, no. 1, pp. 1–8, 2016.

[13] A. Canziani, A. Paszke, and E. Culurciello, "An analysis of deep neural network models for practical applications," *arXiv preprint arXiv:1605.07678*, 2016.

[14] N. Abbas and Y. Chibani, "Svm-dsmt combination for the simultaneous verification of off-line and on-line handwritten signatures," *International Journal of Computational Intelligence and Applications*, vol. 11, no. 03, p. 1250018, 2012.

[15] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construction and Building Materials*, vol. 157, pp. 322–330, 2017.

[16] 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, pp. 815–823.

[17] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *Journal of Machine Learning Research*, vol. 10, no. Feb, pp. 207–244, 2009.

[18] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.