Q1:

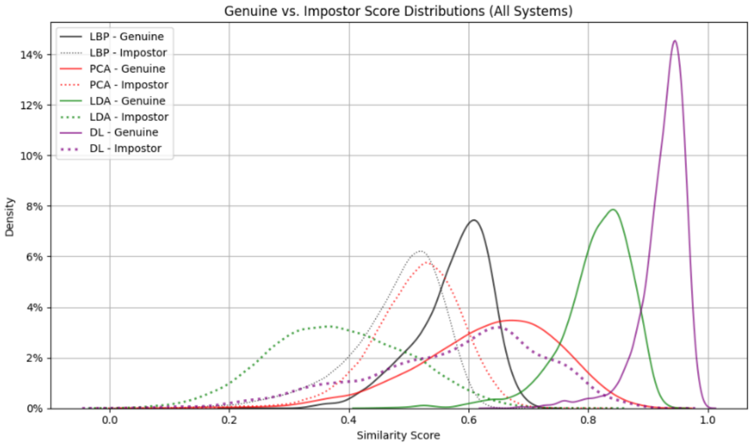
Given the function *dist_metric* and the vector representations in variable *embedded*, the pairwise distances are computed. This is done for PCA, LDA, and the deep learning (DL) embeddings from siamese network using euclidean distance, and for LBP using chi-square distance. The default settings are applied for each model, with the latent representation dimension (*num_components*) set to 35 for PCA, LDA, and DL, and a radius of 1 used for LBP. The distances are subsequently transformed into similarity scores for each model as follows:

$$\text{similarity score}_{\text{i}} = 1\text{-}\frac{\text{distance}_{\text{i}}}{\max\left(\text{distance}_{\text{i}}\right)} ,$$

where i corresponds to the scores and distances from a given model. The similarity score yields values between 0 and 1 - lower distances correspond to higher similarity scores, indicating a higher likelihood of being genuine. These similarity scores are analysed further in Q2.

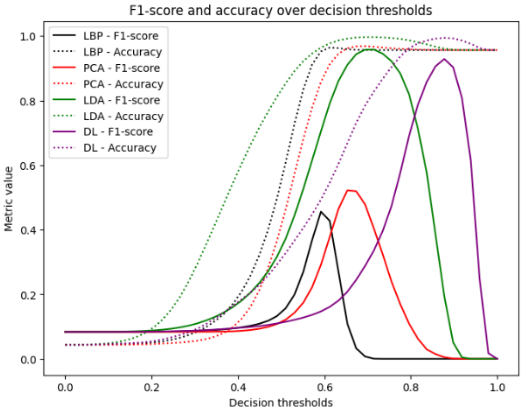
Q2:

The similarity score distributions are normalised by dividing the occurrence of each score by the total number of observations within its respective class, resulting in a probability distribution where the area under the curve is equal to 1. This ensures that the score distributions of two classes - genuine and impostor - are comparable between each other. Normalization addresses class imbalance, allowing for a meaningful comparison between the distributions.



Based on the plot, the LDA model performs the best, showing the smallest overlap between the genuine and impostor score distributions corresponding to the uncertainty interval. This allows the decision threshold to be set so that a low sum of False Rejection Rate (FRR) and False Acceptance Rate (FAR) is achieved. Nevertheless, it is important to assess the trade-offs between specificity and sensitivity for all models. In contrast, the LBP model performs worst, exhibiting the largest overlap. This leads to the highest potential number of combined false non-matches and false matches after thresholding. The DL model achieves a small overlap and is the second best, while the PCA model shows a relatively large overlap, though smaller than that of the LBP model.

Q3:

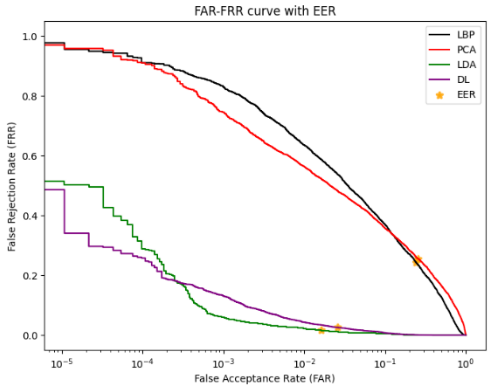


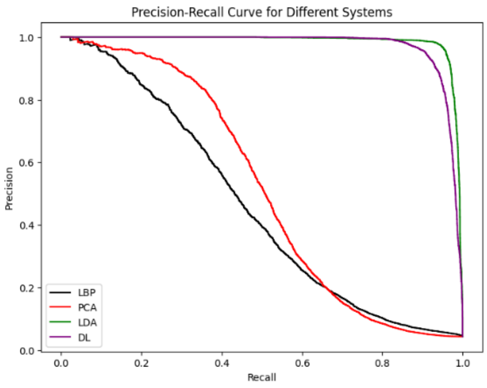
F1 and accuracy scores for a range of thresholds are plotted. To obtain the potential optimal threshold, the results of maximising the F1 score and accuracy are obtained with the corresponding thresholds and their respective metric values:

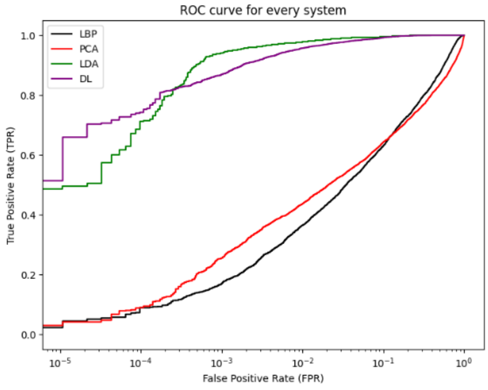
Model	Max F1-Score	Threshold (F1 max)	Accuracy (F1 max)	Max Accuracy	Threshold (Acc max)	F1-Score (Acc max)
LBP	0.4558	**0.5918**	0.9522	0.9643	0.6122	0.4275
PCA	0.5219	**0.6531**	0.9621	0.9685	0.6939	0.4788
LDA	0.9581	**0.7143**	0.9964	0.9964	0.7143	0.9581
DL	0.9288	**0.8776**	0.9939	0.9939	0.8776	0.9288

The F1 score is an effective metric for evaluating models when both high precision and recall are required, and these two factors should be given equal weighting. The LDA model performs best, achieving the highest F1-score of 0.9581, meaning that it can provide high precision and recall at a decision threshold of 0.7143. At the same threshold, it also achieves maximum accuracy - therefore, we derive this as an optimal threshold for the LDA model. The second-best model is the DL model, which achieves a high maximal F1-score of 0.9288 at a threshold of 0.8776. At the same threshold, it also achieves maximum accuracy - therefore, we derive this as an optimal threshold for the DL model. PCA and LBP perform poorly, achieving much lower maximal F1-scores as well as lower maximal accuracy. Also, the thresholds differ for these models depending on which metric is maximised. However, accuracy is biased towards the majority class, while the F1-score is not. Therefore, we consider the F1-score to be a better metric with which to determine an optimal threshold here, provided it does not result in a significant reduction in accuracy. The LBP model performs worst, achieving the lowest maximal F1-score of 0.4558 at the decision threshold of 0.5918. For this threshold, it achieves comparable accuracy to its maximal accuracy, so we determine 0.5918 as the optimal decision threshold for LBP. The PCA model performs slightly better, achieving a maximal F1-score of 0.5219 at a decision threshold of 0.6531, achieving comparable accuracy to its maximal accuracy, so we determine 0.6531 as the optimal decision threshold for PCA.

Q4:







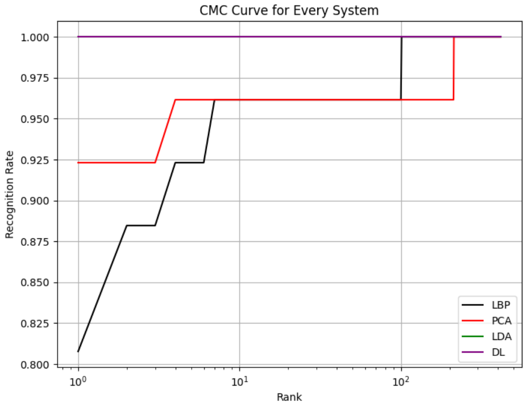
Model	EER Value	ROC AUC	Average Precision
LBP	0.2432	84.01%	0.4604
PCA	0.2557	81.14%	0.5176
LDA	0.0162	99.85%	0.9853
DL	0.0260	99.71%	0.9724

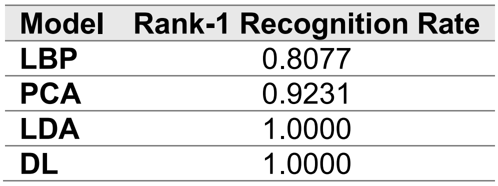
A lower EER indicates a better system, as it means that both types of errors—incorrectly accepting impostors and incorrectly rejecting genuine users—are less frequent when equal. Thus, the LDA model performs best with an EER of 0.0162. The DL model performs slightly worse as the second best, with an EER of 0.026. PCA performs the worst, achieving an EER of 0.2557. LBP performs only slightly better, achieving an EER of 0.2432. EER is a good operating point to evaluate and compare models if the goal is to have equally good systems in terms of incorrectly accepting an unauthorized user and incorrectly rejecting an authorized user. However, if we prefer either security or convenience, it is important to further analyze the ROC curve.

Regarding the ROC curve, it can be observed that LDA achieves the highest True Positive Rate (TPR) for most of the given False Positive Rate (FPR) values. This means that for most decision thresholds, LDA performs best among all the systems. However, at very low FPR values, DL outperforms LDA and the other systems by achieving higher TPR values. This indicates that at low FPR values, DL accepts more genuine cases than LDA, making it a better choice for highly secure biometric systems that require low FPR with relatively stricter decision thresholds. As the FPR increases and the decision threshold becomes more liberal, LDA outperforms all other systems, indicating it is the best for more convenient systems. LDA also achieves the highest ROC AUC of 99.85%, confirming it as the best model overall. DL is the second best overall, achieving a slightly lower ROC AUC of 99.71%. Furthermore, the PCA model is the worst system overall, achieving the lowest ROC AUC of 81.14%, but it performs better than LBP for more secure systems. LBP, with an ROC AUC of 84.01%, performs slightly better overall than PCA and is better than PCA for more convenient systems..

DL achieves the highest precision at lower recall levels. It is therefore possible to set a relatively stricter decision threshold for DL to have a model with high precision and lower recall, which may accept fewer genuine users but ensures that most of its predicted labels are correct when compared to the ground truth, minimizing false positives and thus ensuring higher precision. Consequently, as suggested earlier, DL may be suitable as a very secure system (e.g., banking application authentication). LDA is the best choice for a convenient system, as it achieves the highest precision at high recall values. LDA is also the best overall, achieving the highest average precision of 0.9853. DL is the second best overall, achieving an average precision of 0.9724. LBP performs the poorest overall, with an average precision of 0.4604, but it achieves higher precision at higher recall levels compared to PCA, making it better for convenient systems. The PCA model is only slightly better overall than LBP, achieving an average precision of 0.5176, and is better than LBP for more secure system.

Q5:





Model	Rank-1 Recognition Rate
LBP	0.8077
PCA	0.9231
LDA	1.0000
DL	1.0000

The Cumulative Match Characteristic (CMC) curve plots the probability that a correct identification is returned within the top-x highest ranked matching scores of a sample. Given the data, the x-axis of CMC curve ranges from 1 to 440, as the number of samples is 440, representing the top-k ranks, and the y-axis represents the Recognition Rate, which is the average fraction of probes where the genuine match is found within the top-k ranks. The Rank-1 Recognition Rate represents the value on the CMC curve where rank equals 1. This metric indicates that the LDA and DL models perform the best, being able to detect 100% of genuine users within the highest score. PCA achieves 92.31%, which is also a good result but slightly worse. LBP performs the poorest, achieving an 80.77% Rank-1 Recognition Rate. As rank increases, both the PCA and LBP models clearly accept more and more users, being able to achieve above 95% recognition rate beyond rank 10.

Q6:

We performed an exhaustive grid search over the parameters. For PCA, LDA and DL using siamese networks, we assess the performance of using different dimensions of latent representations (num_components) over 10, 15, 20, 30, 35 (the default setting), 40, 50 and 100, using ROC-AUC and average precision. For LBP, we assess the performance of using different radii ranging from 1 to 5 by 1, using the same metrics. The other parameters of the models remain as provided.

Model	Configuration	ROC AUC	Average Precision (AP)
LBP	radius=1 (default)	0.8401	0.4604
	radius=2	0.9050	0.7198
	radius=3	**0.9114**	**0.7600**
	radius=4	0.9022	0.7468
	radius=5	0.8914	0.7214
LDA	num_components=10	0.9948	0.9506
	num_components=20	0.9982	0.9844
	num_components=30	**0.9985**	**0.9853**
	num_components=35 (default)	0.9985	0.9853
	num_components=40	0.9985	0.9853
	num_components=50	0.9985	0.9853
	num_components=100	0.9985	0.9853
PCA	num_components=10	0.8253	0.3817
	num_components=20	**0.8609**	**0.5717**
	num_components=30	0.8340	0.5411
	num_components=35 (default)	0.8114	0.5176
	num_components=40	0.7896	0.4773
	num_components=50	0.7545	0.4319
	num_components=100	0.6451	0.2409
DL	num_components=10	0.9719	0.6780
	num_components=20	0.9881	0.8662
	num_components=30	0.9924	0.9393
	num_components=35 (default)	**0.9971**	**0.9724**
	num_components=40	0.9914	0.9164
	num_components=50	0.9940	0.9396
	num_components=100	0.9924	0.9470

For LBP, increasing the radius from 1 to 3 improves both the ROC AUC and the average precision (AP), peaking at a radius of 3 with an AUC of 0.9114 and an AP of 0.76. Beyond this, performance deteriorates slightly for radii of 4 and 5. This suggests that a moderate neighbourhood size of radius = 3 is optimal for LBP, providing the best trade-off between capturing local texture and avoiding the noise associated with larger radii. The default setting is clearly suboptimal, as significantly improved performance is achieved by tuning the radius.

LDA still achieves the best performance, with the highest ROC AUC of 0.9985 and the highest AP of 0.9853 from 30 components onwards. The saturation of performance beyond 30 dimensions indicates minimal risk of overfitting when increasing the number of components within this range. However, we choose 30 as the optimal number of components, as this already provides the best results (the same as for the default setting).

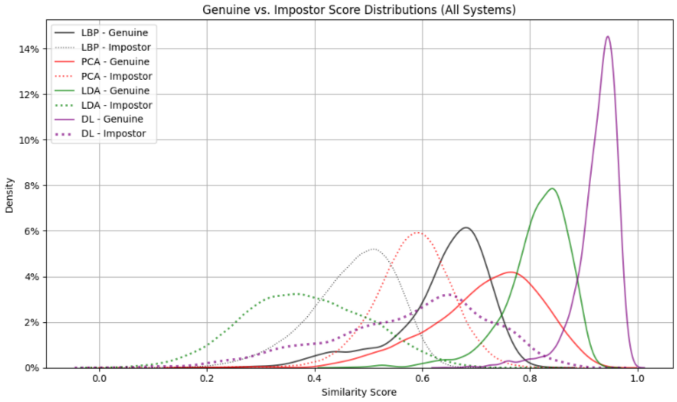
PCA achieves the best performance with 20 components, with an ROC AUC of 0.8609 and an AP of 0.5717. Performance then degrades as the number of components increases to 100. This is because adding too many components introduces noise and leads to overfitting.

Conversely, too few components may underfit. We set the optimal number of components to 20 as this achieved the best performance.

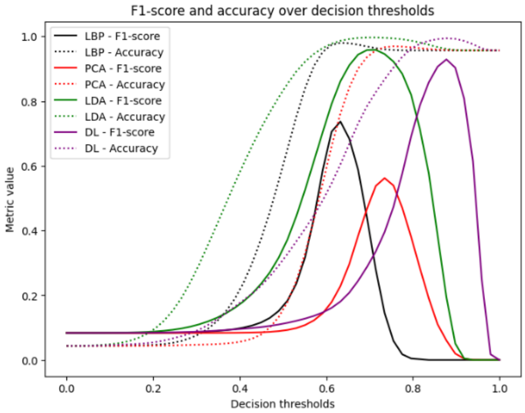
DL using a siamese network shows substantial improvement with an increasing number of components up to 35 (the default), with peak performance of an AUC of 0.9971 and an AP of 0.9724. Beyond this, adding further components leads to a slight drop in both AUC and AP, though the results fluctuate more over the range of component numbers. DL benefits from moderately high dimensional embeddings, but does not require extreme dimensionality to generalise well enough with limited dimensions. Therefore, we maintain 35 as the optimal number of components.

Q7:

The comparison among models under optimal hyperparameter settings, as determined in Q6, is performed using the evaluation metrics from Q2–Q5. Improvements are observed for the LBP and PCA models. The results and conclusions for the DL and LDA models remain unchanged.

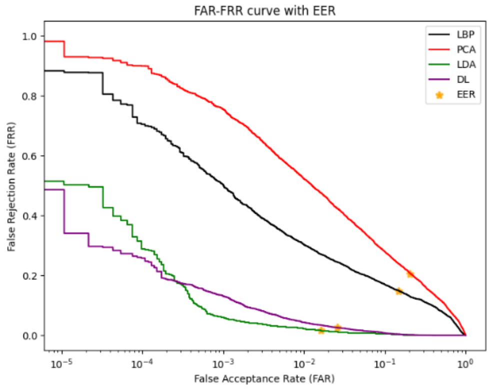


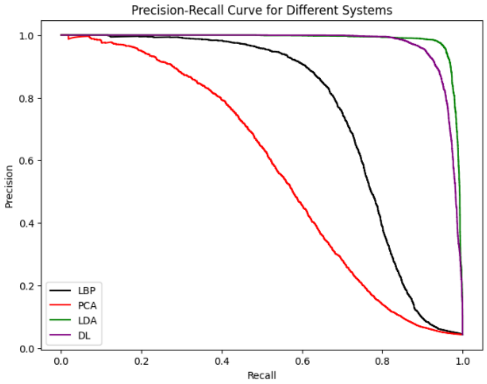
Based on the plot, the LDA model still performs the best, showing the smallest overlap between the genuine and impostor score distributions. DL remains the second best, with visually lower overlap than both LBP and PCA. The LBP model shows significant improvement, as the overlap between the genuine and impostor distributions is now much lower compared to the result with the default setting. In contrast, the PCA model now shows the highest overlap, indicating it as the poorest performing model with the highest potential combined rate of false non-matches and false matches after thresholding. Thus, the improvement for PCA is limited, whereas the improvement for LBP is significant.

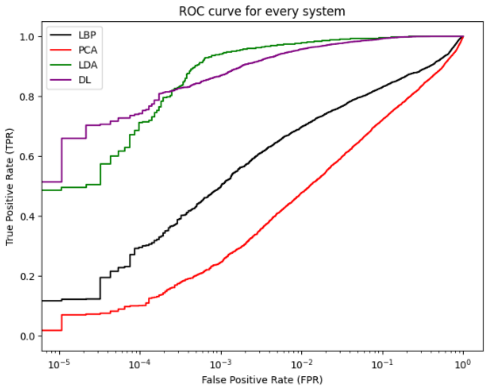


Model	Max F1-Score	Threshold (Max F1)	Accuracy (Max F1)	Max Accuracy	Threshold (Max Acc)	F1-Score (Max Acc)
LBP	0.7367	0.6327	0.9799	0.9799	0.6327	0.7367
PCA	0.5619	0.7347	0.9663	0.9693	0.7551	0.5392
LDA	0.9581	0.7143	0.9964	0.9964	0.7143	0.9581
DL	0.9288	0.8776	0.9939	0.9939	0.8776	0.9288

The LDA model remains the best, with DL as the second best, achieving the two highest maximal F1-scores and accuracies. The LBP model shows significant improvement, achieving a much higher maximal F1-score of 0.7367 at a decision threshold of 0.6327 compared to the previous maximal F1-score of 0.4558. At the threshold of 0.6327, it also achieves its maximum accuracy, making it the optimal threshold for the improved LBP model. PCA now performs worse than LBP, achieving the poorest results with the smallest maximal F1-score and accuracy. So, there is only a limited improvement for PCA, with the maximal F1-score increasing from 0.5219 to 0.5619 at the decision threshold of 0.7347.

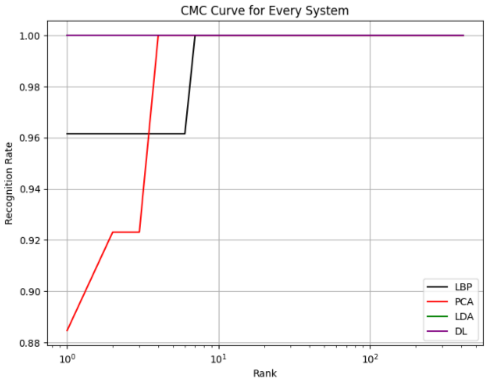






Model	EER Value	ROC AUC	Average Precision
LBP	0.1490	91.14%	0.7600
PCA	0.2061	86.07%	0.5713
LDA	0.0162	99.85%	0.9853
DL	0.0260	99.71%	0.9724

The LDA model remains the best overall in terms of EER, ROC AUC, and AP, with DL as the second best model overall. The conclusions for these two models remain unchanged. The LBP model now outperforms the PCA model in all three overall metrics. It also achieves a higher TPR at every FPR level and higher precision at every recall level, making it the better model for any application compared to PCA. LBP shows significant improvement, with a much lower EER of 0.149, higher ROC AUC of 91.14%, and higher AP of 0.76. Although less substantial, PCA also shows an improvement, with EER of 0.2061, ROC AUC of 86.07% and AP of 0.5713.



Model	Rank-1 Recognition Rate
LBP	0.9615
PCA	0.8846
LDA	1.0000
DL	1.0000

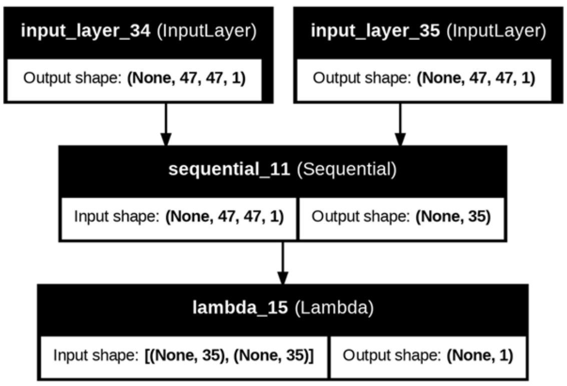
The LBP model shows a significant improvement in CMC-curve performance, achieving a rank-1 recognition rate of 96.15%, higher than PCA's 88.46%. Although PCA's rank-1 recognition rate is lower than before, both LBP and PCA models converge to a 100% recognition rate beyond rank 10, indicating overall improvement.

For additional question 6. is selected

6.

The implemented deep learning model is a convolutional neural network that uses Additive Angular Margin Loss, or ArcFace loss. Unlike a siamese neural network, which takes pairs of images as input, the ArcFace CNN model takes only a single image per forward pass, along with its class label, and learns to classify it into the correct identity using an embedding space. While a siamese network operates on image pairs using a contrastive loss minimising the distance between embeddings of the same identity while maximizing it for different identities, the ArcFace model uses class labels and introduces an angular margin penalty during training to enforce a separation between classes in the embedding space.

Siamese Network


input_layer_34 (InputLayer)
Output shape: (None, 47, 47, 1)
input_layer_35 (InputLayer)
Output shape: (None, 47, 47, 1)
sequential_11 (Sequential)
Input shape: (None, 47, 47, 1)
Output shape: (None, 35)
lambda_15 (Lambda)
Input shape: [(None, 35), (None, 35)]
Output shape: (None, 1)

CNN with ArcFace loss

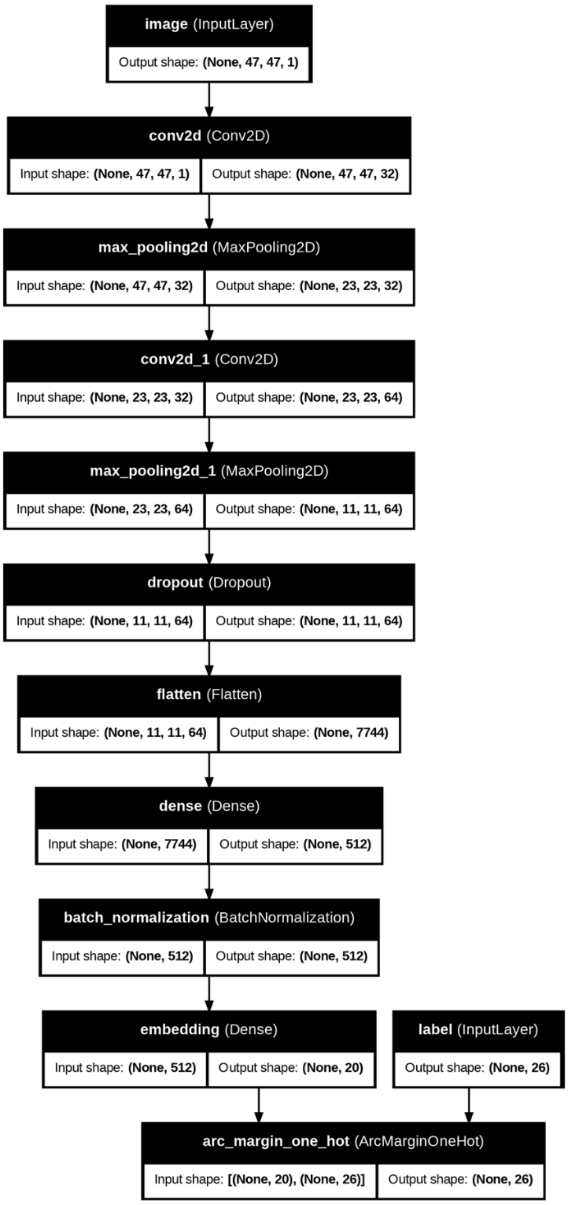


image (InputLayer)
Output shape: (None, 47, 47, 1)
conv2d (Conv2D)
Input shape: (None, 47, 47, 1)
Output shape: (None, 47, 47, 32)
max_pooling2d (MaxPooling2D)
Input shape: (None, 47, 47, 32)
Output shape: (None, 23, 23, 32)
conv2d_1 (Conv2D)
Input shape: (None, 23, 23, 32)
Output shape: (None, 23, 23, 64)
max_pooling2d_1 (MaxPooling2D)
Input shape: (None, 23, 23, 64)
Output shape: (None, 11, 11, 64)
dropout (Dropout)
Input shape: (None, 11, 11, 64)
Output shape: (None, 11, 11, 64)
flatten (Flatten)
Input shape: (None, 11, 11, 64)
Output shape: (None, 7744)
dense (Dense)
Input shape: (None, 7744)
Output shape: (None, 512)
batch_normalization (BatchNormalization)
Input shape: (None, 512)
Output shape: (None, 512)
embedding (Dense)
Input shape: (None, 512)
Output shape: (None, 20)
label (InputLayer)
Output shape: (None, 26)
arc_margin_one_hot (ArcMarginOneHot)
Input shape: [(None, 20), (None, 26)]
Output shape: (None, 26)

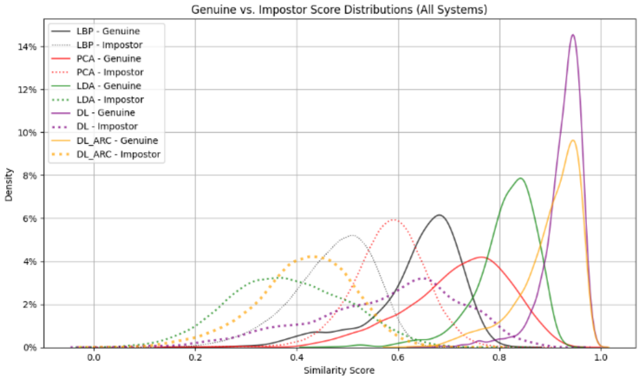
ArcFace applies a similarity learning mechanism that enables metric learning within a classification task by replacing the standard softmax loss with angular margin loss. This is because the softmax loss function does not explicitly optimize the feature embeddings to enforce high intra-class similarity and strong inter-class separation, leading to suboptimal performance in deep face recognition tasks compared to ArcFace. In ArcFace, the cosine similarity is computed between normalized feature embeddings and class weights, followed by the addition of an angular margin to simultaneously enhance intra-class compactness and inter-class discrepancy before scaling the logits. The ArcFace loss is defined as follows:

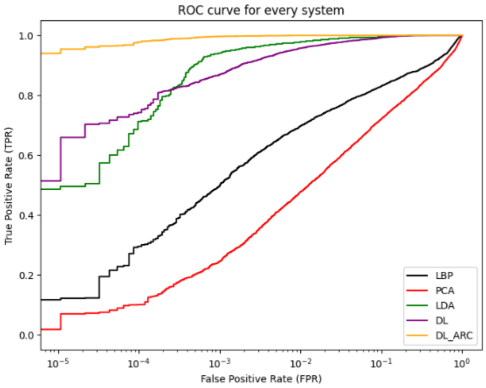
$$\text{Arcface Loss}=\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{s*(\cos(\theta_{y_i}+m))}}{e^{s*(\cos(\theta_{y_i}+m))}+\sum_{j=1,j\neq y_i}^{n}e^{s*\cos\theta_j}},$$

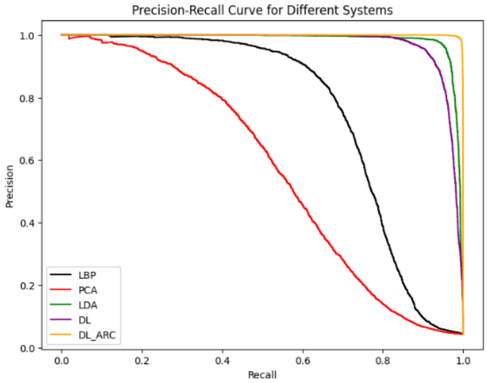
where θ_j is the angle between the weight W_j and the embedding feature x_i, s is feature scale, and m is angular margin penalty. The weights W_j and the embedding feature vectors x_i are l_2 normalized and re-scaled to s. The angular margin m is then added to the angle between the feature and the corresponding class weight to enforce the desired margin.

The model architecture is shown above. It consists of convolutional layers for feature extraction, followed by dense layers that project the features into a 20-dimensional embedding space, which is considered sufficient since this dimensionality achieved satisfactory results for the siamese network, as shown in Q6. The final classification logits are produced by the ArcFace loss layer, enabling the model to learn embeddings that are highly discriminative.

The models are compared using the evaluation metrics from Q2–Q5, including CNN with ArcFace loss, abbreviated as DL_ARC:







Model	EER Value	ROC AUC	Average Precision	Rank-1 Recognition Rate
LBP	0.1490	91.1406%	0.7600	0.9615
PCA	0.2061	86.0737%	0.5713	0.8846
LDA	0.0162	99.8497%	0.9853	1.0000
DL	0.0260	99.7134%	0.9724	1.0000
DL_ARC	**0.0022**	**99.9978%**	**0.9996**	**1.0000**

It can be observed that the newly implemented deep learning model of a deep CNN with ArcFace loss achieves superior performance compared to all other tuned models across every evaluation metric. It exhibits the lowest overlap between genuine and impostor score distributions, the lowest EER, and the highest ROC AUC and average precision, demonstrating the best overall performance. It also achieves higher precision at every recall level and the highest TPR at every FPR level, making it the most effective model for every use case.

However, it is important to note that all models were evaluated on the same dataset they were trained on, as the exercise description emphasized to use embeddings over the entire dataset to evaluate. A valuable extension of this work would be to evaluate models on a separate, held-out test set that was not seen during training. For the siamese network, a proper train/validation/test split was used, although the final evaluation here was still performed on the full dataset. Despite this, the validation loss indicated that the model did not suffer significantly from overfitting. The same applies to the CNN ArcFace model - it was trained with a validation set and achieved consistently low validation loss, suggesting good generalization. Also, since evaluation on whole sample included those validation samples, the models' strong performance further suggests that they are learning good and generalisable representations.