This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TBIOM.2020.2998912, IEEE Transactions on Biometrics, Behavior, and Identity Science

A2-LINK: Recognizing Disguised Faces via Active Learning and Adversarial Noise based Inter-Domain Knowledge

Anshuman Suri, Mayank Vatsa, *Senior Member, IEEE,* and Richa Singh, *Senior Member, IEEE*

Abstract—Face recognition in the unconstrained environment is an ongoing research challenge. Although several covariates of face recognition such as pose and low resolution have received significant attention, “disguise” is considered an onerous covariate of face recognition. One of the primary reasons for this is the scarcity of large and representative labeled databases, along with the lack of algorithms that work well for multiple covariates in such environments. In order to address the problem of face recognition in the presence of disguise, the paper proposes an active learning framework termed as A2-LINK. Starting with a face recognition machine-learning model, A2-LINK intelligently selects training samples from the target domain to be labeled and, using hybrid noises such as adversarial noise, fine-tunes a model that works well both in the presence and absence of disguise. Experimental results demonstrate the effectiveness and generalization of the proposed framework on the DFW and DFW2019 datasets with state-of-the-art deep learning featurization models such as LCSSE, ArcFace, and DenseNet.

Index Terms—Face Recognition, Disguised Faces in the Wild, Impersonation, Obfuscation, Plastic Surgery, Face Verification, Active Learning, Domain Adaptation, Deep Learning

1 Introduction

State-of-the-art face recognition models have demonstrated near-human performance on constrained and semi-constrained datasets such as CMU MultiPIE [1] and Labeled Faces in the Wild [2]. These datasets lack the presence of rich covariates like disguise, makeup, and low-resolution, which are generally present in face images sampled from real-world scenarios. This absence of covariates in the training database leads to models performing poorly in the presence of such covariates during testing. For example, a person may get a photo clicked while wearing sunglasses, a wig, or with makeup. Such disguised appearances should not confuse an ideal face recognition model. At the same time, as shown in Fig. 1, an impostor wearing a disguise to impersonate another user should not be able to fool an ideal face recognition model. Differentiating between these two cases without compromising the performance on normal data is a non-trivial task. This covariate has significant implications for real-world face recognition systems used by government agencies, online social networks, and surveillance systems.

Even when the problem of adapting to new covariates in datasets may be solved, it usually requires a lot of labeled data for training. Obtaining labels for normal face-images, either generated by humans or state-of-the-art algorithms, is a relatively easy task. However, identifying people with makeup and impostors, for instance, is a non-trivial task: even humans need to cross-check with various sources and use agreement with other judges to label such face images confidently. An ideal algorithm should be able to work with small amounts of such labeled data to boost its performance in the presence of covariates. Of all the methods in the literature designed to work with a scarcity of labeled data, active learning seems to be the most suitable approach when labeled data is scarce. The algorithm should be able to select unlabeled examples for which it wishes to obtain labels, instead of asking judges to annotate all of them.

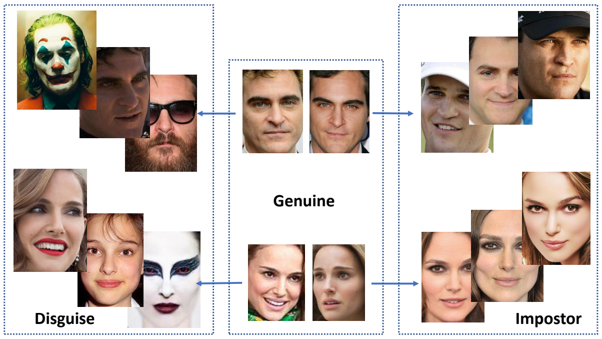


Fig. 1: Image samples of two subjects from the DFW dataset [3], [4] database, along with their corresponding impostor and disguised images.

Data augmentation is a well-known technique that helps increase dataset size and thus helps lower overfitting tendencies when training machine-learning models [5]. Combining various data augmentation methods, along with active learning, should thus help us bridge the performance gap between limited and abundant labeled data with covariates. The central supporting hypothesis behind the design of the proposed algorithm is that using the right kind of data augmentation, not just to augment the dataset size but iden-

- *A. Suri is with Department of Computer Science, SEAS, University of Virginia, Charlottesville, VA, 22904 (e-mail:anshuman@virginia.edu)*
- *M. Vatsa and R. Singh are with Indian Institute of Technology (IIT) Jodhpur, India (email: mvatsa@iitj.ac.in, richa@iitj.ac.in)*

tifying useful data points, can help adapt a model to work within the presence of covariates, all the while consuming lower labeled data than vanilla learning algorithms do. This paper proposes A2-LINK, an active learning algorithm that can be used to train models to work well in the presence of several covariates. A2-LINK utilizes concepts from active learning, domain adaptation, and noise-based augmentation to train models that approach state-of-the-art on multiple datasets while using a fraction of the labeled data.

1.1 Related Work

The following section discusses the development of research in the domain of face recognition in the presence of disguises. Since the proposed algorithm utilizes concepts from domain adaptation and active learning, a brief literature review of these areas is provided in the subsequent section.

1.1.1 Disguises in Face Recognition

Unlike traditional face recognition literature, disguise as a covariate of face recognition has received limited attention. A psychological study on observers' face identification capabilities when presented with images with "disguise" shows how memory performance deteriorates with these disguises, and that not all forms of disguise hinder performance equally [6]. Conclusions from this study validate empirical observations in the performance of face identification and verification models: it is a hard task for humans and thus is an even harder task for artificial cognitive systems. Singh *et al.* proposed a face recognition algorithm that is designed to be robust to changes in appearance such as disguises, and works well with limited gallery images [7]. Ramanathan *et al.* proposed a framework that compensates for pose variations using 'half-faces' to derive a similarity measure to be robust to changes in age, disguise, illumination, and pose [8]. Although these techniques achieved then state-of-the-art performance, there have been considerable advancements in performance with the advent of sophisticated deep learning algorithms. Li *et al.* proposed a robust face recognition system that uses low-resolution 3D cameras for identification [9]. Other work explores the feasibility of face verification under disguise variations, using multi-spectrum face images: combining visible, near, and mid-visible spectral imaging cameras that can better identify disguises like makeup and surgical alterations [10].

Anti-spoofing techniques have also been explored to increase robustness to disguises [11], [12]. Other techniques distinguish between biometric (regions without disguise) and non-biometric (regions with disguise) facial features using visible and thermal face images for better face recognition under disguise [13]. However, these methods require thermal face images or 3D face maps, which are nearly impossible to obtain from cameras used in standard equipment like drones, mobiles, and CCTVs. To circumvent this limitation, Jiang *et al.*propose a 3D face alignment pipeline that extracts 3D faces using 2D face images at inference time to extract face landmarks efficiently [14]. Their algorithm shows promising results in handling disguise using 3D face models to remove disguise from images. However, the performance of this approach, while better than baseline methods, is not near state-of-the art for the dataset. It is computationally intensive compared to a standard deep learning model, making it unsuitable for real-time deployment. Smirnov *et al.* describe an approach based on hard example mining to impose a useful structure in their mini-batches, which is used by the training algorithm to train models on a dataset with the disguise covariate [15]. They also propose doppelganger mining: exploiting similarities between subjects for imposing a structure on batches when training models [16]. Although these works show promising results, they may not necessarily work well in the absence of sufficient labeled data, which significantly limits their performance in the presence of rare covariates.

The DFW workshop organized at CVPR in 2018 released a new dataset with disguises: the Disguised Faces in the Wild (DFW) dataset [4], [17], including a large set of face images with various forms of disguise. More recently, a bigger, more diverse version of the DFW dataset was recently made public at the DFW 2019 workshop organized at ICCV 2019: the DFW 2019 dataset [18]. These datasets have helped advance research in disguised face recognition [11], [19], [20], [21], [22], [23]. For instance, Deng *et al.* use novel face detection and alignment algorithms, along with ArcFace [24] face feature embeddings to handle disguises [25], and are currently state-of-the-art on DFW2019. However, their algorithm is not conservative in the amount of data it needs to train, and thus cannot be extended to scenarios where labeled disguised data is scarce.

1.1.2 Domain Adaptation and Active Learning

Matching faces with their disguised variants can be modeled as a domain adaptation problem: the source domain contains undisguised faces (data without covariates), whereas the target domain may include disguised images (data with covariates). Recent work by Kan *et al.* [26] uses such an approach to utilize unlabeled data in the target domain. However, their work cannot function on top of the already trained models, or convolutional neural networks/deep learning models, which is what most effective face recognition systems use. Yao *et al.* utilize a similar approach: they consider low resolution and high resolution as two distinct domains and propose a projection technique that utilizes domain adaptation [27] to project data from one domain to the other. The scarcity of labeled data from the target domain can significantly lower the performance of such approaches. One possible solution to this scarcity is to use weakly-supervised learning to use model-generated labels. However, the magnitude of disguise in most cases can be too much for a model trained on the source domain to work well. Even for humans, it can be difficult to tell apart between real actors and doppelgangers[1]!

Active learning, a technique that intelligently selects examples while training or queries the annotator (Oracle) for labels, has demonstrated promising results in environments constrained by the availability of labeled data. The DFAL algorithm by Ducoffe *et al.* estimates distances of points from decision boundaries using adversarial perturbations [28], and then order those data samples according to their usefulness. Generative Adversarial Networks (GANs) have been used for active learning as well: the generator generates

[1] http://iab-rubric.org/DFW/index.html

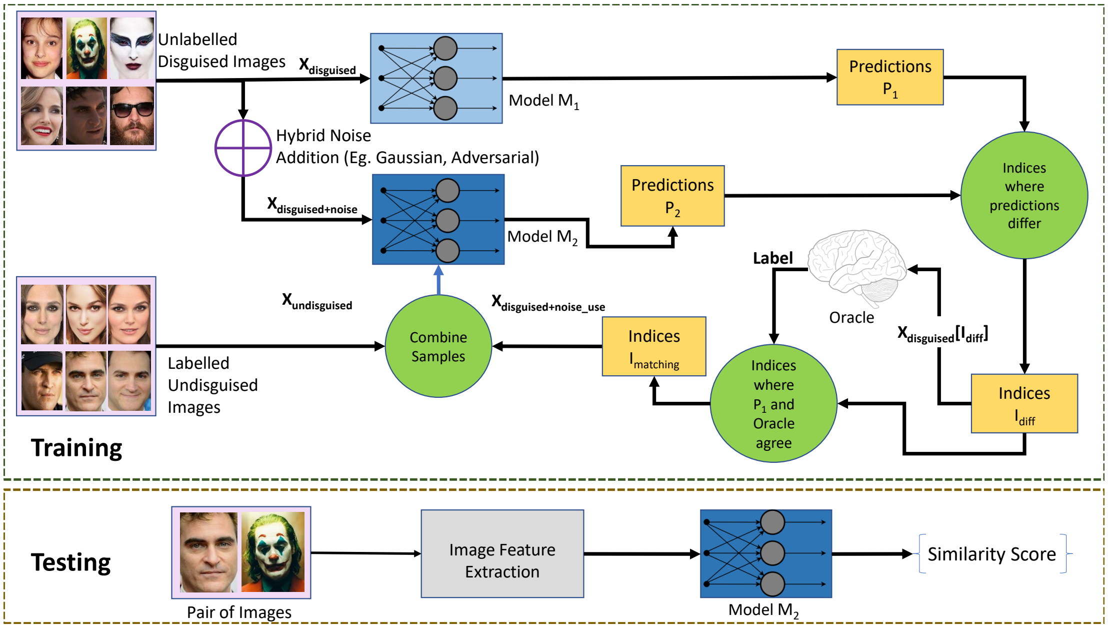


Fig. 2: A data-flow diagram of A2-LINK for the case of disguised faces. A2-LINK starts with a batch of unlabeled data from the target domain and ends up with a batch of data from both source and target domain, on which M_2 is fine-tuned.

images which, if labeled, can maximize information gain for the model [29]. Another example of adapting active learning to deep learning is work by Geifman *et al.* that uses intermediate-layer activations to sample data points to be labeled [30].

Combining active learning with domain adaptation can, potentially, lead to a system that works better than either of the techniques applied individually: a face recognition system that is robust to covariates, while not requiring a lot of labeled training data with covariates. A combination of these two techniques has been briefly explored in literature, one example of which is active-supervised domain adaptation [31]. This algorithm trains an auxiliary classifier to segregate examples into the source or target domain. However, it may not be feasible to draw such a strict distinction in the case of face disguises. Additionally, an auxiliary classifier increases both the time needed to train and run inference via this algorithm.

1.2 Research Contributions

Inspired by the recent success of domain adaptation and active learning, this paper proposes A2-LINK (***A****dversarial Noise and* ***A****ctive-****L****earning based* ***I****nter-domai****N*** ***K****nowledge*): a framework that utilizes active learning to adapt to data with covariates. The key contributions of this paper are:

- Propose a framework that fine-tunes a model trained on a small set of labeled data with covariates, to improve its performance in the presence of covariates. This paper builds on top of A-LINK[2] [32] and introduces an adversarial noise component while constructing hybrid noise inputs for the algorithm.
- Empirically demonstrates the generalizability of A2-LINK to various featurization models, namely: DenseNet [33], Local Class Sparsity Supervised Autoencoder (L-CSSE) [34], and ArcFace [24].
- Evaluate the performance of models trained with A2-LINK on all protocols of the DFW dataset [4], [17] and DFW2019 dataset [18]. The proposed algorithm achieves close to state-of-the-art performance while using a fraction of the labeled data.

2 A2-LINK Algorithm

The algorithm's novelty lies in determining which points from the target distribution should be used to fine-tune a model that already performs well on data in the absence of covariates. A2-LINK intelligently creates a subset of images sampled from the target domain (data with covariate), which in effect allows the best possible transfer of weights from the source to the target domain. Fig. 2 illustrates the concept of A2-LINK for the case of disguise as a covariate. The algorithm assumes the following entities:

- A model M_1 (**teacher model**): having knowledge of the source domain, via being trained well on data without covariates.
- A model M_2 (**student model**): with some partial information about the target domain, via being trained on a relatively smaller set of labeled data with covariates.
- An *Oracle*: an annotator which **yields** the corresponding **ground truth** for given inputs. This entity is found in most active learning settings; its use in this particular problem is elaborated upon in Section 2.4.

For both models M_1 and M_2, they take as input a data point (which, in this paper, is a pair of face images) and output a confidence score $p \in [0, 1]$.

$$p = M(x) \tag{1}$$

A predicted value of 0 implies that the images in a pair are of distinct people, while 1 corresponds to a perfect match.

[2] A shorter version of the manuscript was presented at IEEE International Conference on BTAS, 2019 [32].

A threshold can be set on this predicted score to classify two images as matching or different people based on the desired precision-recall trade-off of the parties using these trained models. The ultimate goal of the algorithm is to use M_1's knowledge to refine M_2's performance on the target domain in a semi-supervised manner.

The presence of two models helps achieve an effect similar to model distillation [35], transferring knowledge from M_1 to M_2. As discussed in Section 2.4, the presence of an Oracle ensures that the data used to fine-tune M_2 does not have incorrect or noisy labels, thus ensuring that these outliers do not hamper the model's performance. The algorithm can be broken down into four phases:

1) Generating predictions from both models M_1 and M_2,
2) Filtering data based on some heuristics,
3) Querying the Oracle to obtain ground truth for selected samples, and
4) Fine-tuning M_2.

These four steps are repeated in a loop. These steps are described in detail in the following sections, along with details on how the final trained model M_2 is used at inference time for generating predictions on test data.

2.1 Generating Predictions

The main objective of the algorithm is to fine-tune M_2 with a limited amount of labeled training data, by utilizing domain information available from M_1. Since data annotation is an expensive process, this paper considers the setting where a sufficient amount of unlabeled data is available as the starting premise for the algorithm.

The algorithm samples a batch of **unlabeled** target-domain image-pairs (called X_{target}) and passes it to M_1 to obtain a set of predictions, P_1:

$$P_1 = M_1(X_{target}) \tag{2}$$

X_{target} is then partitioned into n disjoint subsets:

$$\begin{aligned} X_{target} &= \bigcup_{i=1}^{n} X_{target}^{i} \\ X_{target}^{i} \cap X_{target}^{j} &= \phi \quad \forall i, j \end{aligned} \tag{3}$$

and noise is added to a copy of this batch:

$$X_{target+noise} = \bigcup_{i=1}^{n} \{\epsilon_i(x) \mid x \in X_{target}^{i}\} \tag{4}$$

where ϵ_i is a particular noise function, that produces a data point with noise $\epsilon_i(x)$, for the given input data point x.

There is no restriction on how many noises or what kind of noise (noise function) can be incorporated into the image; the noise function may be purely statistical, or even adversarial. In this paper, both these variants are explored: Gaussian, Salt and Pepper, Perlin, Speckle, and Poisson noise, as well as Pixel-wise perturbations [36]. Various combinations of all these noises are also explored.

2.1.1 Adversarial Noise

Adversarial noise is imperceptible noise which, when added to a data point, can fool a predictive model with high confidence. Generating adversarial examples for a Siamese network has not been studied much in literature. Although it is straightforward to do so in a black-box setting where the adversary has control over only one image (and the gallery image cannot be accessed or modified), crafting white-box examples is a non-trivial task. This paper works with Pixel-wise perturbations [36], as it is a black-box attack and can be used with other kinds of machine-learning models like Random Forests. To craft adversarial examples, the algorithm concatenates both image inputs to the network and adds pixel-wise noise to this combined image. Once the combined images have been perturbed, the algorithm splits them back into two images to be used by the Siamese network.

The noise crafted above is independent of the model's specific architecture. Model-specific adversarial perturbations like FGSM [37], or a framework similar to AMC [38] could potentially be used to generate various adversarial noise components. However, most of such attacks are white-box attacks and thus assume the target models to be from a specific family of classification models (neural networks). Using black-box, gradient-free adversarial perturbations helps keep the overall algorithm model-agnostic and computational costs of generating perturbed inputs independent of the model's complexity, which helps scale the algorithm with larger models.

It may be noted that for Adversarial noise components, model access is also needed to craft adversarial examples. Thus, the noise crafted in Eq. (4) case would be computed as $\epsilon_i(x, M_1)$.

2.2 Predictions from M_2

This set of now-noisy samples ($\mathrm{X}_{target+noise}$) is fed as input to M_2 to obtain P_2:

$$P_2 = M_2(X_{target+noise}) \tag{5}$$

The intuition here for adding noise is: a model that has overfitted on the given data distribution or is not confident enough about its predictions is most likely to yield incorrect predictions for such perturbed inputs. However, if the model predicts a score that is close to the actual label (which is approximated using P_1) for a pair of images perturbed with noise, it is expected to perform well for unperturbed images as well. Thus, adding noise to images and considering only such cases, helps identify data points which, when used to fine-tune M_2, could potentially improve its performance.

2.3 Data Filtering

Once P_1 and P_2 are obtained, the algorithm computes the set of indices I_{diff} as the indices where P_1 and P_2 differ in predictions:

$$I_{diff} = \{i \mid \mathbb{I}[P_1[i] \geq 0.5] \neq \mathbb{I}[P_2[i] \geq 0.5]\} \tag{6}$$

where $\mathbb{I}$ denotes the indicator function. Similarly, this paper defines agreement for predictions P_1 and P_2 corresponding to data point with index i, if:

$$\mathbb{I}[P_1[i] \geq 0.5] = \mathbb{I}[P_2[i] \geq 0.5] \tag{7}$$

```
Algorithm 1: A2-LINK
Input: mix_ratio, $n$ hybrid noise functions $\epsilon_i$
Train model $M_1$ on image-pairs without the covariate (source domain);
Train model $M_2$ on the limited-size set of image-pairs with the covariate (target domain);
while $X_{target}$ contains data do
    Get next batch of unlabeled image-pairs with covariate ($X_{target}$);
    $P_1$ = $M_1(X_{target})$;
    Partition $X_{target}$ into $n$ subsets $X^i_{target}$;
    Collect noise-added samples: $X_{target+noise}$ = $\bigcup_{i=1}^{n} \{ \epsilon_i(x) \mid x \in X^i_{target} \}$;
    // When $\epsilon_i$ is Adversarial, use $\epsilon_i(x, M_1)$
    $P_2$ = $M_2(X_{target+noise})$;
    $I_{diff} = \{i \mid \mathbb{I}[P_1[i] \geq 0.5] \neq \mathbb{I}[P_2[i] \geq 0.5]\}$ ;
    $I_{diff}$ = $\{i \mid P_1[i] < 0.5 - \epsilon\} \cup \{i \mid P_1[i] > 0.5 + \epsilon\}$;
    $Label$ = $\{ O(X_{target}[i]) \mid i \in I_{diff} \}$;
    $I_{matching} = \{i \in I_{diff} \mid Label[i] = P_1[i]\}$ ;
    $X_{target+noise\_use}$ = $X_{target+noise}[I_{matching}]$;
    Get $X_{source}$ = $mix\_ratio - 1$ more batches of source-domain image-pairs;
    Fine-tune $M_2$ with $concatenate(X_{source}, X_{target+noise\_use})$ and their corresponding labels;
end
```

The intuition here is that data points, for which P$_1$ and P$_2$ agree, would not provide any additional information required to fine-tune M$_2$. Thus, using such data-points would not be very beneficial in improving M_2's performance, and querying the Oracle to confirm their labels would not be very useful, given a specific budget of label queries.

To further reduce the number of queries made to the Oracle, the algorithm filters I$_{diff}$ the following way:

$$I_{diff} = \{i \mid P_1[i] < 0.5 - \epsilon\} \cup \{i \mid P_1[i] > 0.5 + \epsilon\} \quad (8)$$

Discarding points for which the predictions from M_1 lie in [0.5-ϵ, 0.5+ϵ] helps lower the number of queries made to the Oracle, since these cases correspond to the scenario where M_1 is not confident enough about its predictions and thus have a higher probability of being incorrect.

2.4 Using the Oracle

After the first filtering step, the algorithm utilizes the Oracle to prune data points further. The Oracle provides the ground truth for any input it is supplied with: the Oracle ascertains whether a pair of images belong to the same class/identity or not (elaborated in Section 4).

Once I$_{diff}$ is computed, the corresponding image-pairs, *i.e.* X$_{target}$[I$_{diff}$] are collected and queried to the Oracle to obtain ground-truth as:

$$Label = \{O(X_{target}[i]) \mid i \in I_{diff}\} \quad (9)$$

where $O(x)$ is the ground truth corresponding to input data x. In the same way, as explained in Section 2.3, the algorithm chooses points where the Oracle agrees with M_1's predictions, to generate a new set of indices I$_{matching}$:

$$I_{matching} = \{i \mid Label[i] = \mathbb{I}[P_1[i] \geq 0.5], i \in I_{diff}\} \quad (10)$$

Sections 2.3 and 2.4 primarily comprise the **active-learning** portion of the proposed algorithm: the algorithm queries the Oracle only for a small subset of possible image-pairs since such label queries are expensive in the real world.

2.5 Fine-tuning M_2

After computing $I_{matching}$ from the previous step, image pairs from X_{target} corresponding to these indices are selected:

$$X_{target+noise\_use} = \{X_{target+noise}[i] \mid i \in I_{matching}\} \quad (11)$$

These image-pairs correspond to the cases where M_1 is right about its predictions but M_2 is wrong about its predictions when a perturbed version of the same data-points is used.

Since this batch consists of only data from the target domain (with covariate), M_2 is susceptible to overfitting on image-pairs from the target domain. In order to circumvent such a possibility, some data from the source domain (with associated labels) is added to the batch as well: $mix\_ratio - 1$ more batches of data-points from the source domain to prepare one final batch on which M_2 is fine-tuned.

$$\begin{aligned} X_{source} &= mix\_ratio - 1 \text{ batches from source domain} \\ X_{use} &= concatenate(X_{source}, X_{target+noise\_use}) \quad (12) \end{aligned}$$

This step ensures that M_2 maintains performance on the source domain, while it is being fine-tuned on data from the target domain. The parameter mix_ratio can be used to control how much M_2 is supposed to change. A high ratio can be used when data from the target domain is expected to be rare at inference time, whereas a one-to-one ratio ($mix\_ratio = 2$) can be used when data from both the domains are equally probable. The algorithm repeats the above four subsections in order, looping through batches of unlabeled data from the target domain. The entirety of the algorithm is given in Algorithm 1.

2.6 Testing

After the A2-LINK algorithm has completed, we have the model M_2 that can be used for inference on data from both the source domain and the target domain. The model predicts a score $\in [0, 1]$, where a higher score suggests an identity match between the two input images, and a score of 0 indicates a mismatch. A gallery of images (one or more images per identity) is used as a reference for identifying people in the input image. Given an input image, images from the gallery are tested one at a time together with the input image, and M_2 yields a score for each such pair. Then, the identity from the gallery whose corresponding pair had the highest score is finalized as the identity matching the given input image. This procedure is fairly standard when working with Siamese networks.

3 Datasets

This paper evaluates the effectiveness of the proposed algorithm on two popular disguise databases: DFW dataset [4], [17] and DFW2019 dataset [18]. The details of both the databases are described below. A comparison of the DFW and the DFW2019 datasets, highlighting their fundamental differences, is summarized in Table 1.

3.1 Disguised Faces in the Wild (DFW)

The DFW dataset [4], [17] contains 11,157 images of 1000 subjects with different kinds of disguise variations. As per the predefined protocol, 400 subjects comprise the training set, and 600 subjects comprise the testing set. Face coordinates for these images are included in the dataset and were generated using Faster RCNN [39]. The face region from each image is extracted using these coordinates. The dataset contains images with their identifiers. However, for training on these protocols, the proposed algorithm requires a format where a pair of images with a $\{0, 1\}$ label (an indicator of them being the same) is provided. Thus, image-pairs are constructed by combining inter-class and intra-class images (all possible combinations) for use in the algorithm. Subject exclusivity is maintained between the training and test sets while creating these image-pairs. All further references to *data point* refer to these image-pairs, not individual images from the original structure of the dataset. The three cases or protocols considered for evaluation are:

1) **Impersonation**: considering genuine validation (595 cases) vs. impostor impersonators (24,451 cases).
2) **Obfuscation**: considering genuine, disguised (13,302 cases) vs. cross-subject impostors (9,027,981 cases).
3) **Overall**: considering genuine (disguised and undisguised both; 13,897 cases) vs. impostors (impersonators and cross-subject; 9,052,432 cases).

3.2 Disguised Faces in the Wild 2019 (DFW2019)

The DFW2019 dataset [18] is meant to be a dataset for evaluation, built along the lines of the original DFW dataset. It contains 3840 images of 600 subjects with new disguise variations like bridal makeup and plastic surgery. Similar to the procedure used for DFW dataset, image-pairs are constructed by combining inter-class and intra-class images. The dataset defines four protocols for evaluation:

1) **Impersonation**: considering genuine validation (250 cases) vs. impostor impersonators (7,431 cases).
2) **Obfuscation**: considering genuine, disguised (along with bridal makeup) (10,267 cases) vs. cross-subject impostors (2,802,011 cases).
3) **Plastic Surgery**: considering genuine, before-after surgery (250 cases) vs. cross-subject impostors (124,500 cases).
4) **Overall**: considering genuine (disguised and undisguised both; 10,767 cases) vs. impostors (impersonators and cross-subject; 2,933,942 cases).

4 Implementation Details

Models: To assess the generalization of the proposed approach to different feature extraction models, this paper considers experiments with three featurization models: Local Class Sparsity Supervised Autoencoder (L-CSSE) [34], ArcFace [24] (Section 5.4), and DenseNet [33] (Section 5):

- The L-CSSE model utilizes class-specific sparsity patterns in the latent space to train an autoencoder. An L-CSSE model pre-trained on the LFW database [2] is used in this paper.
- Densenet uses a convolutional model where each layer is connected to every other layer in a feed-forward fashion. Similar to L-CSSE, a DenseNet model pre-trained on the LFW database [2] is used in this paper.
- ArcFace utilizes the geometry of data to use geodesics on the latent space manifold. A pre-trained ArcFace model, pre-trained on a modified version of the MS-Celeb-1M dataset [40] is used in this paper.

Attribute	DFW	DFW2019
Training Set	3386 images, 400 subjects	-
Testing Set	7771 images, 600 subjects	3840 images, 600 subjects
Covariates	Disguise, Makeup (shades, beards, etc)	Bridal Makeup, Plastic Surgery, etc
Evaluation Protocols	Impersonation, Obfuscation, Overall	Impersonation, Obfuscation, Plastic Surgery, Overall
Challenges	Peculiar covariates, Unconstrained disguises	Peculiar covariates, No training set

TABLE 1: A comparison of the DFW and DFW2019 datasets.

The featurization model is followed by a Siamese network built on top of it with three fully-connected layers: the absolute difference in feature vectors is passed as input to the fully-connected layers. All feature extraction layers are frozen while training these Siamese networks.

Architectures of models M_1 and M_2 are the same: it consists of an absolute difference layer over the two inputs (features extracted from images), followed by two layers with ReLU [41] activation of 512 and 64 neurons. A two-neuron layer with Sigmoid activation follows these layers, thus predicting a score in the range [0,1]. The AdaDelta optimizer, with its default learning rate of 1, was used while training all the models, with a batch size of 64. M_1 is trained using *labeled, undisguised* face-image pairs, while M_2 is trained using 50% of the *labeled, disguised* face-image pairs available. The proposed algorithm uses the remaining 50% disguised face-image pairs for A2-LINK. All cropped face-images are resized to 224×224. Since DFW2019 dataset does not contain a training set, models trained on DFW dataset are used to evaluate the performance on DFW2019 dataset.

Code: Nvidia K-80 GPU with 128GB RAM and a Xeon E5-2630v2 CPU is used to perform all the experiments. Tensorflow[3] and Keras[4] were used for implementing the algorithm (MXNet[5] for ArcFace) and training models[6]. While training on both the datasets, Disparity-ratio is varied in the range $\{0.25, 0.5, 1, 2, 4\}$. An Oracle is **artificially simulated** by accessing ground-truth for labeled data from the target domain, which is otherwise not used by the algorithm, and keeping track of the number of such accesses.

[3] https://www.tensorflow.org/
[4] https://keras.io/
[5] https://mxnet.apache.org/
[6] https://github.com/iamgroot42/A-LINK

For generating prediction matrices needed for running evaluation on various dataset protocols (DFW, DFW2019 datasets), for all $\binom{n}{2}$ image-pairs, (for n people) predictions are obtained by passing image-pairs through the model. For evaluation, this generated predictions matrix is used along with a masking matrix provided with the datasets.

Hyper-parameters: For different kinds of noise utilized, the following configurations were used:

- Gaussian: $\mu = 10, \sigma = 10$
- Salt & Pepper: Salt/Pepper ratio 0.5, Amount = 0.004
- Speckle: $\mu = 0$, $\sigma = \frac{1}{15}$
- Pixel-wise perturbations: pixel count = 40, iterations = 50, population size = 250

A grid search is performed over the hyper-parameters specified in Algorithm 1. The configuration of parameters that yield the best results is as follows:

- $\epsilon = 0.05$
- mix_ratio: 2
- 50% of labeled disguised-face data used in Step 2
- noise used: Gaussian, Salt-Pepper, Poisson, Speckle, Perlin, Adversarial, and their combination.

5 Experimental Results and Analysis

This section presents the performance of A2-LINK on the DFW and DFW2019 databases. As defined in the DFW competition protocols, the results are reported as Genuine accept rates (GAR) at two different false accept rates (FAR). Figs 3 and 4, along with Tables 2-4 show results of A2-LINK trained with DenseNet, LCSEE, and ArcFace models on the DFW and DFW2019 datasets.

In addition to the models trained with A2-LINK, results for model M_1 and M_2 (M_2 **before A2-LINK**) after the step where it has been trained on a limited-size set of disguised face-image-pairs (Line 2, Algorithm 1) are also included. Measuring the increase in performance from this model to the final model helps quantify performance gains when using A2-LINK (i.e. ablative study).

5.1 Performance on DFW Dataset

Fig. 3 and Tables 2-3 summarize the results on the DFW database. Compared to the base model, each of the featurization model shows significant improvement with the proposed algorithm. For the configuration using DenseNet (Section 4):

- Use of A2-LINK leads to absolute improvements of 6.53%, 4.87%, and 5.02% in GAR at 1%FAR for the cases of impersonation, obfuscation, and overall case of DFW dataset, respectively.
- Similarly, absolute improvements of 5.31%, 6.17%, and 6.86% are observed in GAR at 0.1%FAR for the cases of impersonation, obfuscation, and overall case of DFW dataset, respectively.

For comparison, results from the DFW competition organized with CVPR 2018 are also included. According to the competition, MiRA-Face [17], AEFRL [15] and UMDNets [42] are the current state-of-the-art for this dataset. It can be observed that in addition to outperforming base models, the performance of models trained with A2-LINK

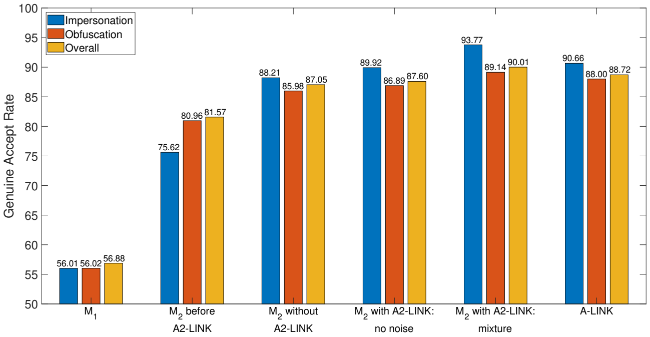


Fig. 3: GAR at 1% FAR for impersonation, obfuscation, and overall performance on DFW dataset, for M_1 and variations of M_2. M_1, M_2 use L-CSSE for feature extraction.

TABLE 2: GAR at 1% FAR and 0.1% FAR for impersonation, obfuscation and overall performance on the DFW dataset. Featurization model: DenseNet.

Model	$GAR_{1\%}$	$GAR_{0.1\%}$
Impersonation		
Baseline (VGG-Face)	52.77	27.05
Baseline (VGG-Face2)	73.94	38.48
AEFRL [15]	96.80	57.64
MiRA-Face [17]	95.46	51.09
UMDNets [42]	94.28	53.27
A-LINK [32]	95.73	75.38
M_1 (DenseNet)	89.68	65.60
Uncertainty Sampling [43]	89.71	65.78
Margin Sampling [44]	91.09	73.12
Entropy Sampling [45]	91.27	73.53
M_2 before A2-LINK	89.15	69.41
M_2 without A2-LINK	91.38	71.93
M_2 after A2-LINK: no noise	92.84	73.27
M_2 after A2-LINK: mixture	**97.91**	**77.24**
Obfuscation		
Baseline (VGG-Face)	31.52	15.72
Baseline (VGG-Face2)	54.86	31.55
MiRA-Face [17]	90.65	**80.56**
AEFRL [15]	87.82	77.06
UMDNets [42]	86.62	74.69
A-LINK [32]	88.97	72.13
M_1 (DenseNet)	83.11	63.01
Uncertainty Sampling [43]	83.44	63.28
Margin Sampling [44]	85.01	68.92
Entropy Sampling [45]	85.07	68.99
M_2 before A2-LINK	84.23	65.15
M_2 without A2-LINK	86.99	68.95
M_2 after A2-LINK: no noise	87.52	69.28
M_2 after A2-LINK: mixture	**91.86**	75.12
Overall		
Baseline (VGG-Face)	33.76	17.73
Baseline (VGG-Face2)	56.22	32.68
MiRA-Face [17]	90.62	**79.26**
AEFRL [15]	87.90	75.54
UMDNets [42]	86.75	72.90
A-LINK [32]	89.30	72.72
M_1 (DenseNet)	83.74	63.18
Uncertainty Sampling [43]	83.89	63.71
Margin Sampling [44]	85.50	65.97
Entropy Sampling [45]	86.08	69.04
M_2 before A2-LINK	85.41	65.99
M_2 without A2-LINK	87.56	69.53
M_2 after A2-LINK: no noise	88.14	70.15
M_2 after A2-LINK: mixture	**92.58**	76.39

TABLE 3: GAR at 1% FAR and 0.1% FAR for impersonation, obfuscation and overall performance on the DFW dataset. Featurization model: ArcFace.

Model	$GAR_{1\%}$	$GAR_{0.1\%}$
Impersonation		
M_1 (ArcFace) [24]	98.66	60.84
A-LINK [32] (with ArcFace)	98.80	62.50
M_2 after A2-LINK: mixture noise	**99.01**	**69.27**
Obfuscation		
M_1 (ArcFace) [24]	95.08	92.20
A-LINK [32] (with ArcFace)	95.42	92.59
M_2 after A2-LINK: mixture noise	**95.93**	**93.08**
Overall		
M_1 (ArcFace) [24]	95.11	91.76
A-LINK [32] (with ArcFace)	95.50	92.14
M_2 after A2-LINK: mixture noise	**95.99**	**93.01**

TABLE 4: GAR at 0.1% FAR and 0.01% FAR for impersonation, obfuscation, plastic surgery, and overall performance on the DFW2019 dataset. Featurization model: ArcFace.

Model	$GAR_{0.1\%}$	$GAR_{0.01\%}$
Impersonation		
Baseline (LightCNN)	74.40	51.20
ArcFaceIntraInter [24]	56.80	17.60
M_1 (ArcFace)	72.40	44.80
A-LINK [32] (with ArcFace)	76.40	52.80
M_2 before A2-LINK	72.00	42.80
M_2 without A2-LINK	76.40	51.20
M_2 after A2-LINK: no noise	77.60	52.80
M_2 after A2-LINK: mixture	**79.20**	**54.40**
Obfuscation		
Baseline (LightCNN)	55.56	36.90
ArcFaceIntraInter [24]	98.92	**98.43**
M_1 (ArcFace)	95.73	91.43
A-LINK [32] (with ArcFace)	96.84	94.02
M_2 before A2-LINK	94.48	90.70
M_2 without A2-LINK	96.14	92.50
M_2 after A2-LINK: no noise	97.03	94.14
M_2 after A2-LINK: mixture	**99.00**	97.20
Plastic Surgery		
Baseline (LightCNN)	69.20	47.20
ArcFaceIntraInter [24]	98.40	95.60
M_1 (ArcFace)	94.80	87.60
A-LINK [32] (with ArcFace)	95.20	92.00
M_2 before A2-LINK	90.40	88.80
M_2 without A2-LINK	91.20	89.60
M_2 after A2-LINK: no noise	95.20	92.00
M_2 after A2-LINK: mixture	**98.80**	**96.00**
Overall		
Baseline (LightCNN-29v2)	55.74	36.50
ArcFaceIntraInter [24]	98.45	93.64
M_1 (ArcFace)	95.29	88.86
A-LINK [32] (with ArcFace)	95.96	93.06
M_2 before A2-LINK	93.89	88.01
M_2 without A2-LINK	95.74	90.38
M_2 after A2-LINK: no noise	96.90	93.26
M_2 after A2-LINK: mixture	**98.63**	**96.18**

is at par with the current state-of-the-art (Table 2). Compared to A-LINK [32], an average absolute increase of 2.78% for GAR at 1% FAR, and 4.73% for GAR at 0.1% FAR is observed. These observations imply that adding adversarial noise to the hybrid collection of noise components significantly helps the algorithm's performance. Overall, as shown in Table 3, the best performance of over 93% GAR at 0.1% FAR is observed using ArcFace as the base model.

5.2 Exploring Variations of A2-LINK on DFW Database

Some of the steps in Algorithm 1 can be replaced with other variations, making the approach generic to the presence of any covariates in the data:

- Although using 50% of the available data to initially train M_2 gives the best results, varying this ratio in the range {30, 40, 50}% does not alter the results significantly. For instance, with 30% data, the drop in the performance was only 2.0% and with 40% data, the drop is about 1.1%. Thus, one can start with M_2 trained on a smaller number of labeled examples from the target domain without compromising much on the model's performance when fine-tuned with A2-LINK.
- Experiments with model agnostic noises (Step 7, Algorithm 1) including Gaussian, Salt and Pepper, Speckle, Poisson, Perlin, and a mixture of these were performed. This paper also considers adversarial noise: specifically Pixel-wise perturbations. A mixture of all these noises is observed to perform best on the DFW datasets.
- Steps 10 and 13 (Algorithm 1) check for equality by considering the two outputs as part of a binary classification problem. This criterion can be replaced with:

  $$I_{diff} = argsort(- \mid P_1 - P_2 \mid)[: sample\_size] \quad (13)$$

  $sample\_size$ is a hyper-parameter and can be set as a specific percentage of $\mid P_1 - P_2 \mid$ (**disparity-ratio**). This ratio is varied in {0%, 12.5%, 25%, 50%}. With the normal criteria defined in Section 2.3 (i.e., not use the rule described above) yields the best results. With disparity-ratio of 12.5%, the proposed model yields about 1.4% reduced performance, whereas with 50% disparity-ratio, the difference increases to about 8%. Higher disparity ratio leads to a stricter selection criterion for shortlisting data points for fine-tuning, which may lead to the algorithm missing out on potentially useful data.
- Experiments with varying the values of ϵ (Step 11 of Algorithm 1) are performed. A higher value of ϵ corresponds to selecting more extreme cases, which leads to fewer queries to the Oracle. At the same time, it results in less data for fine-tuning M_2. ϵ is varied in {0, 0.05, 0.10}, with $\epsilon = 0.05$ giving the best results, where the other two values yield at least 3% lower recognition performance.
- Some active-learning techniques are explored as well[7]:
  Uncertainty Sampling selects data instances based on usefulness; 1 - maximum posterior probability (across classes). Yang *et al.*utilize this in their proposed algorithm for diversity maximization [46].
  Margin Sampling: selects data instances based on the difference between the highest and second-highest posterior probabilities (across classes). Wu *et al.*propose a weighted sampling method for deep embeddings [47].
  Entropy Sampling: selects data instances based on Shannon entropy over all the classes [48].
  Experiments are performed by varying the upper cap on the number of queries that were made by the active

[7] modAL (https://modal-python.github.io/) is used for implementing these active-learning algorithms.

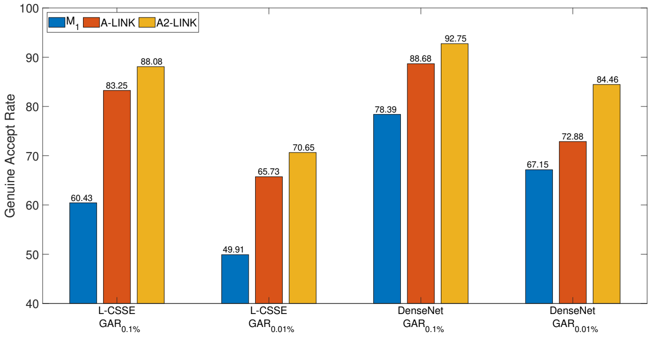


Fig. 4: GAR at 0.1% FAR and 0.01% FAR for Protocol 4 (overall) on DFW2019 dataset, for M_1, A-LINK, and A2-LINK, with featurization models DenseNet and L-CSSE.

learning algorithm. For all the ratios, the proposed algorithm outperforms the three above mentioned active learning techniques (Table 2).

Studying these variations shows that the proposed algorithm is not highly sensitive to specific configurations of hyper-parameters, suggesting that the boost in performance observed across datasets and featurization models is not specific to a particular hyper-parameter setting.

5.3 Performance on DFW2019 Dataset

As shown in Fig. 4, on the DFW2019 database, for the configuration using DenseNet:

- at 0.1% FAR, compared to without using the A2-LINK algorithm, the GAR of the proposed model shows absolute improvements of 2.8%, 2.86%, 7.6%, and 2.89% for the impersonation, obfuscation, plastic surgery, and overall protocols, respectively.
- at 0.01% FAR, compared to without using the A2-LINK algorithm, the GAR of the proposed model shows absolute improvements of 3.2%, 4.7%, 6.4%, and 5.8% for the impersonation, obfuscation, plastic surgery, and overall protocols, respectively.

Results with ArcFace as base model are shown in Table 4 and Fig. 4 shows the results of LCSEE as the base model. For comparison, results from the DFW2019 competition are also included in Table 4. According to the competition, LightCNN-29v2 [49] and variants of ArcFace [24] are the state-of-the-art for this dataset. Trends in performance similar to the case of DFW dataset are observed: A2-LINK outperforms current state-of-the-art for almost all protocols on this dataset (Table 4). Compared to A-LINK [32], an average absolute increase of 2.81% in GAR at 0.1%FAR, and 2.98% for GAR at 0.01% FAR is observed.

5.4 Generalization Over CNN Models

As a proof-of-concept of the proposed approach and its generalization capabilities across featurization models, experiments with three feature extraction models are performed: L-CSSE model, DenseNet, and ArcFace. As can be observed from Figs. 3 and 4, along with Tables 2- 4, training with A2-LINK significantly boosts the performance for all three featurization models. Models trained with A2-LINK even outperform A-LINK [32] by a significant margin, thus reinforcing the importance of including adversarial noise components in the proposed algorithm.

The core contribution of this algorithm lies in the performance gain it yields while being constrained by the amount of labeled training data available. Traditionally trained L-CSSE and DenseNet yield limited performance on these datasets. However, the proposed active-learning based approach enhances their performance significantly, all the while using a fraction of the labeled data.

5.5 Discussion

The proposed algorithm and its previous version (A-LINK [32]) give a reasonably good increase in performance, both when dealing with disguise and multi-resolution as covariates. The core intuition, and thus the driving factor behind the algorithm's superior performance, is to help the model adapt better to changes in the input distribution. When switching from regular to disguised face images, there is a significant domain shift that a vanilla model cannot handle well. By introducing the concept of images with added noise, the algorithm can coarsely simulate the effects of using inputs different from the original input distribution, without actually requiring samples from the distribution with an additional covariate.

It may be argued that the algorithm is close to using pure data augmentation (with an adversarial loss function, if adding adversarial noise) while training the model. However, there are two key differences:

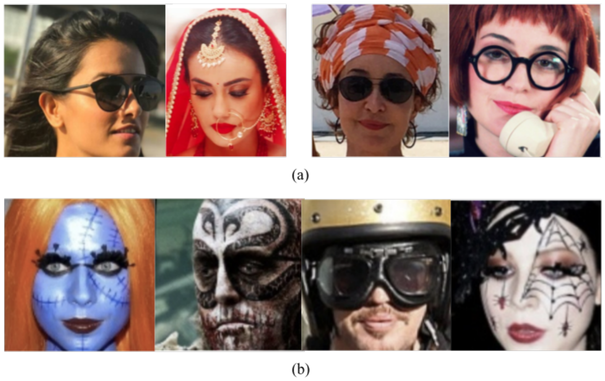
1) Data augmentation augments the training data while training the model. A2-LINK, in contrast, uses passive feedback from the model itself to efficiently sample unlabeled data points from the target domain for labeling, and then adds them for training.
2) Data augmentation incorporates "all" the additional training data, whereas the objective of the proposed A2-LINK is to identify the data points that are useful for learning the models.

These hypotheses are also validated from the results. The results obtained using only data augmentation or only active learning algorithms are quite far from the models trained using A2-LINK in terms of performance.

5.5.1 Analysis

It is observed that, on average, A2-LINK required 30-35% less labeled disguised face images while training the algorithm. For all the protocols of DFW and DFW2019 datasets, a proportionate mixture of Gaussian, Salt-Pepper, Poisson, Speckle, Perlin, and Adversarial noise outperforms individual variations for the noises it uses. Several ablation studies are performed to analyze the importance of individual components of the proposed algorithm:

Noise: In order to assess the importance of noise addition in A2-LINK, the proposed model is compared with the variation that does not add any noise (M_2 **after A2-LINK: no noise**). As expected, the variation of A2-LINK with no noise is outperformed by the one with multiple types of noise (Tables 2, 4). Tuning $mixture\_ratio$ significantly alters the performance: a $1:1$ ratio of noisy disguised images and clean undisguised images yields the best results. Increasing



(a)

(b)

Fig. 5: Success and failure cases of A2-LINK: (a) pairs with correct results where the proposed A2-LINK yields correct results but other algorithms incorrectly classify the samples (first pair is impostor sample and second pair is genuine sample), and (b) challenging probe images where none of the algorithms, including A2-LINK, are able to perform correct classification.

the ratio of noisy images tends to make M_2 overfit on noisy images, whereas increasing the proportion of unperturbed images tends to dilute the effect of images with added noise.

Adversarial Noise: To study the importance of including adversarial noise, A2-LINK is compared with the variant that contains all the noise components except adversarial (A-LINK [32]). For all datasets and featurization models that are considered (Figs 3, 4), Adversarial noise improves the performance significantly compared to A-LINK. These observations imply that the Adversarial noise component is an important part of A2-LINK.

Active sampling: To study the effect of actively selecting samples while fine-tuning M_2, the paper performed experiments on the variation in which M_2 is trained without running A2-LINK; using all available labeled disguised-faces data (M_2 **without A2-LINK**). The proposed model outperforms this variation in all cases (Tables 2, 4 and Fig. 3).

In addition to these ablation studies, we also inspect success and failure cases of the models to get a better understanding of the workings of the models. As shown in Fig 5(a), the proposed A2-LINK algorithm is able to distinguish challenging cases and yields correct classification results. However, DFW databases contain some tough samples with face-paint and accessories which cover almost entire face (as shown in Fig 5(b)) where none of the algorithms, including A2-LINK, are able to match them with their genuine mated pairs correctly.

5.5.2 Future Directions

Although A2-LINK achieves state-of-the-art accuracy on multiple datasets and feature-extraction models, there is significant scope for improvement. Some interesting research directions are:

1) Instead of using the same combination of noise components in each iteration, the algorithm could use a dynamic strategy to estimate which noise components can be most useful for each batch of data. Such a strategy would also scale well if using a large pool of noise components.
2) The proposed algorithm can be extended, with relevant modifications, to other related problem statements such as kinship analysis [50] as well as to other problem domains such as natural language processing.

6 Conclusion

The proposed A2-LINK algorithm combines concepts from active learning, domain adaptation, and hybrid noise augmentation to train models to achieve near state-of-the-art performance while being constrained by the amount of labeled data with covariates available. Further, A2-LINK is faster than training a model on all the data points, has low auxiliary storage requirements, and reduces the number of labeled examples required significantly, without compromising on the model's performance. Experimental results show that A2-LINK leads to significant improvements while fine-tuning a model, for all protocols of the DFW dataset, as well as the DFW2019 dataset. The proposed algorithmic framework shows good generalization: both across featurization models (L-CSSE, Densenet, ArcFace) as well as different covariates: disguise in DFW dataset, and bridal makeup and plastic surgery in DFW2019 dataset. Since A2-LINK is generic, it can be used to incorporate more sophisticated active-learning criteria, along with variations of noise, while fine-tuning the model under consideration.

The ability to train accurate models with limited amounts of labeled data with covariates can be crucial in real-world applications. The proposed algorithm provides an elegant way to perform this task and may be extended for other applications. Furthermore, while the proposed algorithm can be modified to work with multiple, sufficiently different covariates simultaneously, it is an exciting direction to explore.

Acknowledgments

M. Vatsa is supported through the Swarnajayanti Fellowship by the Government of India.

References

[1] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.
[2] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," in *Workshop on faces in 'Real-Life'Images: detection, alignment, and recognition*, 2008.
[3] T. I. Dhamecha, R. Singh, M. Vatsa, and A. Kumar, "Recognizing disguised faces: Human and machine evaluation," *PLOS ONE*, vol. 9, no. 7, pp. 1–16, 07 2014.
[4] M. Singh, R. Singh, M. Vatsa, N. K. Ratha, and R. Chellappa, "Recognizing disguised faces in the wild," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 1, no. 2, pp. 97–108, 2019.
[5] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *arXiv preprint arXiv:1712.04621*, 2017.
[6] G. Righi, J. J. Peissig, and M. J. Tarr, "Recognizing disguised faces," *Visual Cognition*, vol. 20, no. 2, pp. 143–169, 2012.
[7] R. Singh, M. Vatsa, and A. Noore, "Face recognition with disguise and single gallery images," *Image and Vision Computing*, vol. 27, no. 3, pp. 245–257, 2009.

[8] N. Ramanathan, R. Chellappa, and A. R. Chowdhury, "Facial similarity across age, disguise, illumination and pose," in *Proceedings of the International Conference on Image Processing*, vol. 3. IEEE, 2004, pp. 1999–2002.

[9] B. Y. Li, A. S. Mian, W. Liu, and A. Krishna, "Using kinect for face recognition under varying poses, expressions, illumination and disguise," in *Proceedings of the Workshop on Applications of Computer Vision*. IEEE, 2013, pp. 186–192.

[10] I. Pavlidis and P. Symosek, "The imaging issue in an automatic face/disguise detection system," in *Proceedings IEEE Workshop on Computer Vision Beyond the Visible Spectrum: Methods and Applications (Cat. No. PR00640)*. IEEE, 2000, pp. 15–24.

[11] T. Kim, Y. Kim, I. Kim, and D. Kim, "Basn: Enriching feature representation using bipartite auxiliary supervisions for face anti-spoofing," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019.

[12] J. Liu and A. Kumar, "Detecting presentation attacks from 3d face masks under multispectral imaging," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 47–52.

[13] T. I. Dhamecha, A. Nigam, R. Singh, and M. Vatsa, "Disguise detection and face recognition in visible and thermal spectrums," in *Proceedings of the International Conference on Biometrics*, 2013, pp. 1–8.

[14] L. Jiang, X.-J. Wu, and J. Kittler, "Dual attention mobdensenet (damdnet) for robust 3d face alignment," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[15] E. Smirnov, A. Melnikov, A. Oleinik, E. Ivanova, I. Kalinovskiy, and E. Luckyanets, "Hard example mining with auxiliary embeddings," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 37–46.

[16] E. Smirnov, A. Melnikov, S. Novoselov, E. Luckyanets, and G. Lavrentyeva, "Doppelganger mining for face representation learning," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 1916–1923.

[17] V. Kushwaha, M. Singh, R. Singh, M. Vatsa, N. Ratha, and R. Chellappa, "Disguised faces in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 1–9.

[18] M. Singh, M. Chawla, R. Singh, M. Vatsa, and R. Chellappa, "Disguised faces in the wild 2019," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, Oct 2019.

[19] A. Subramaniam, A. Narayanan Sridhar, and A. Mittal, "Feature ensemble networks with re-ranking for recognizing disguised faces in the wild," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, Oct 2019.

[20] Q. Gao, J. Cheng, D. Xie, P. Zhang, W. Xia, and Q. Wang, "Tensor linear regression and its application to color face recognition," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019.

[21] N. Kohli, D. Yadav, and A. Noore, "Face verification with disguise variations via deep disguise recognizer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 17–24.

[22] S. Vishwanath Peri and A. Dhall, "Disguisenet: A contrastive approach for disguised face verification in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 25–31.

[23] S. Suri, A. Sankaran, M. Vatsa, and R. Singh, "On matching faces with alterations due to plastic surgery and disguise," in *Proceedings of the International Conference on Biometrics Theory, Applications and Systems*, 2018, pp. 1–7.

[24] J. Deng, J. Guo, X. Niannan, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4690–4699.

[25] J. Deng and S. Zafeririou, "Arcface for disguised face recognition," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019.

[26] M. Kan, J. Wu, S. Shan, and X. Chen, "Domain adaptation for face recognition: Targetize source domain bridged by common subspace," *International Journal of Computer Vision*, vol. 109, no. 1-2, pp. 94–109, 2014.

[27] Y. Yao, X. Li, Y. Ye, F. Liu, M. K. Ng, Z. Huang, and Y. Zhang, "Low-resolution image categorization via heterogeneous domain adaptation," *Knowledge-Based Systems*, vol. 163, pp. 656–665, 2019.

[28] M. Ducoffe and F. Precioso, "Adversarial active learning for deep networks: a margin based approach," *arXiv preprint arXiv:1802.09841*, 2018.

[29] J.-J. Zhu and J. Bento, "Generative adversarial active learning," *arXiv preprint arXiv:1702.07956*, 2017.

[30] Y. Geifman and R. El-Yaniv, "Deep active learning over the long tail," *arXiv preprint arXiv:1711.00941*, 2017.

[31] A. Saha, P. Rai, H. Daumé, S. Venkatasubramanian, and S. L. DuVall, "Active supervised domain adaptation," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2011, pp. 97–112.

[32] A. Suri, M. Vatsa, and R. Singh, "A-link: Recognizing disguised faces via active learning based inter-domain knowledge," in *Proceedings of the IEEE International Conference on Biometrics: Theory, Applications and Systems*, 2019.

[33] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2017, pp. 2261–2269.

[34] A. Majumdar, R. Singh, and M. Vatsa, "Face verification via class sparsity based supervised encoding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1273–1280, 2017.

[35] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Proceedings of the NIPS Deep Learning and Representation Learning Workshop*, 2015.

[36] J. Su, D. V. Vargas, and K. Sakurai, "One pixel attack for fooling deep neural networks," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 5, pp. 828–841, 2019.

[37] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proceedings of the International Conference on Learning Representations*, 2015.

[38] D. Vijaykeerthy, A. Suri, S. Mehta, and P. Kumaraguru, "Hardening deep neural networks via adversarial model cascades," in *International Joint Conference on Neural Networks (IJCNN)*, 2019.

[39] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Proceedings of the Advances in Neural Information Processing systems*, 2015, pp. 91–99.

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

[41] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning*, 2010, pp. 807–814.

[42] A. Bansal, R. Ranjan, C. D. Castillo, and R. Chellappa, "Deep features for recognizing disguised faces in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 10–16.

[43] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Springer-Verlag New York, Inc., 1994, pp. 3–12.

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

[45] C. E. Shannon, "A mathematical theory of communication," *Bell system technical journal*, vol. 27, no. 3, pp. 379–423, 1948.

[46] Y. Yang, Z. Ma, F. Nie, X. Chang, and A. G. Hauptmann, "Multi-class active learning by uncertainty sampling with diversity maximization," *International Journal of Computer Vision*, vol. 113, no. 2, pp. 113–127, Jun 2015.

[47] C.-Y. Wu, R. Manmatha, A. J. Smola, and P. Krahenbuhl, "Sampling matters in deep embedding learning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2840–2848.

[48] B. Settles, "Active learning literature survey," University of Wisconsin-Madison Department of Computer Sciences, Tech. Rep., 2009.

[49] X. Wu, R. He, Z. Sun, and T. Tan, "A light cnn for deep face representation with noisy labels," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2884–2896, 2018.

[50] N. Kohli, M. Vatsa, R. Singh, A. Noore, and A. Majumdar, "Hierarchical representation learning for kinship verification," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 289–302, 2017.

Anshuman Suri received his B.Tech (Hons) degree in Computer Science from the Indraprastha Institute of Information Technology, Delhi, in 2018 with an undergraduate thesis on “Hardening Deep Neural Networks via Adversarial Model Cascades”. He is currently pursuing his doctoral degree in Computer Science at the University of Virginia, under the guidance of Professor David Evans. His research interests are deep learning and the privacy and security aspects of machine learning, focusing on adversarial robustness.

Mayank Vatsa received the M.S. and Ph.D. degrees in computer science from West Virginia University, USA, in 2005 and 2008, respectively. He is currently a Professor with IIT Jodhpur, India, and the Project Director of the Technology and Innovation Hub on Computer Vision and Augmented & Virtual Reality under the National Mission on Cyber Physical Systems by the Government of India. He is also an Adjunct Professor with IIIT-Delhi, India and West Virginia University, USA. His areas of interest are biometrics, image processing, machine learning, computer vision, and information fusion. He has co-edited books on Deep learning in Biometrics and Domain Adaptation for Visual Understanding. He is the recipient of the Prestigious Swarnajayanti Fellowship from the Government of India, the A. R. Krishnaswamy Faculty Research Fellowship at the IIIT-Delhi, and several best paper and best poster awards at international conferences. He is an Area/Associate Editor of Information Fusion and Pattern Recognition, the General Co-Chair of IJCB 2020, and the PC Co-Chair of IEEE FG2021, ICB 2013, IJCB 2014, and ISBA2017. He has also served as the Vice President (Publications) of the IEEE Biometrics Council where he started the IEEE Transactions on Biometrics, Behavior, And Identity Science.

Richa Singh received the Ph.D. degree in computer science from West Virginia University, Morgantown, USA, in 2008. She is currently a Professor at IIT-Jodhpur, India, and an Adjunct Professor with IIIT-Delhi and West Virginia University, USA. She has co-edited book Deep Learning in Biometrics and has delivered tutorials on deep learning and domain adaptation at ICCV 2017, AFGR 2017, and IJCNN 2017. Her areas of interest are pattern recognition, machine learning, and biometrics. She is a fellow of IAPR and a Senior Member of IEEE and ACM. She was a recipient of the Kusum and Mohandas Pai Faculty Research Fellowship at the IIIT-Delhi, the FAST Award by the Department of Science and Technology, India, and several best paper and best poster awards in international conferences. She has also served as the Program Co-Chair of AFGR2019 and BTAS 2016, and a General Co-Chair of ISBA 2017. She is currently serving as the General Co-Chair of FG2021 and a Program Co-Chair of IJCB 2020. She is also the Vice President (Publications) of the IEEE Biometrics Council. She is an Associate Editor-in-Chief of Pattern Recognition, and Area/Associate Editor of several journals.