**Hierarchical Siamese Triple Ranking Convolution Network in Signature Forgery Detection**

**Solution Design**

Signature fraud or forgery can be broadly of two types:

1. Blind Forgery: The first type of forgery is where the forger has no idea what the signature to be forged looks like. This should be easy to detect by machine because it is usually not very close to the appearance of a genuine signature.
2. Skilled Forgery: The second type of forgery is either simulation or tracing, in which the forger has a sample of the signature to be forged. In this case, detecting fraud requires more sophisticated tools to differentiate minute but critical details between genuine and forged signatures.

We propose a hierarchical framework to detect fraud signatures using Image processing and Deep convolutional Siamese networks wherein we first detect forgeries of the first type (blind forgery), and then subsequently detect skilled forgeries in the second step of the hierarchy.

In the first stage of the hierarchical framework, we plan to implement a *pairwise training* to compute the *dissimilarity score* between each pair of image embeddings of signatures, whether they are intended for the same person or otherwise. So, each pair will have a dissimilarity score, and the label whether they are supposed to be genuine or blind forgery. We model this to using Siamese Convolutional Nets coupled with generalized linear model architecture with logistic loss functions. The images which will be blind forged will be detected in this step and filtered out and the other ones will be sent to the next step.

In the second stage, which is the most critical, we deal with forgeries where the signature looks very similar to the original. Here we propose a *deep triplet ranking network* using Siamese Convolution Network and *triplet loss* as the objective function using *Bayesian optimization* for the model hyperparameters.

The best part about this framework is once the model is trained, we require just one base image to determine whether another signature image is genuine or not with one shot learning.

**Tech Stack**

* Python
* R
* Keras
* Tensorflow
* OpenCV

**Algorithm : Hierarchical Siamese Convolution Network**

**Step 1: Detecting blind forgeries**

In the first step of the hierarchy we contrive the training data in a way where each observation is *pair* of images, either both of a person’s genuine signature, or one of person’s genuine signature, and the other of someone else’s signature (can be true/forged). These will have labels (class) genuine or fraud assigned to them.

The idea here is to determine whether two signature pairs appear to be of the same person or not. Once we construct the training data in this format, we train the model based on convolutional Siamese network. Siamese convolution networks are twin networks with shared weights, which can be trained to learn the feature embeddings where similar observations are placed in proximity and dissimilar are placed apart. This is achieved by exposing the network to a pair of similar and dissimilar observations and minimizing the dissimilarity between similar pairs while simultaneously maximizing it between dissimilar pairs as shown in Fig.1.

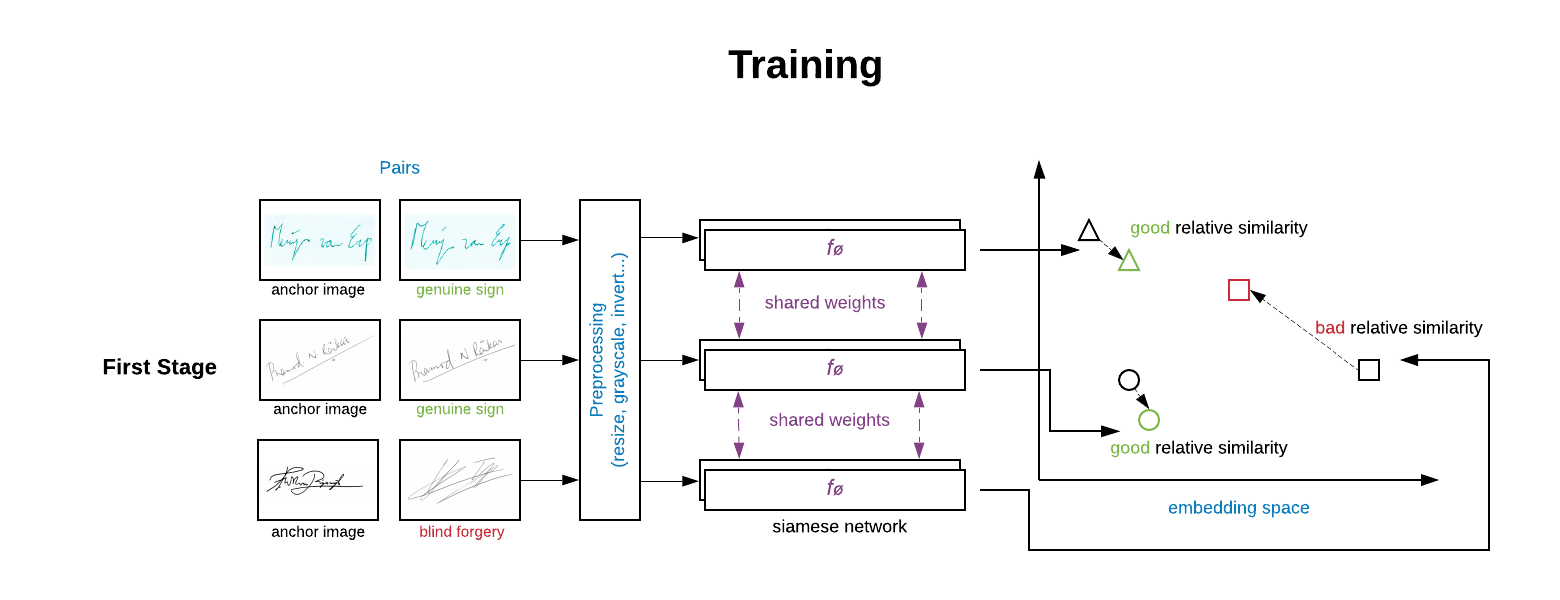


Fig.1: First stage to detect blind forgery using Siamese Twin architecture

Let’s say,

*a & b are the two signature images of different individuals (blind forged).*

*xa , xb be the image embeddings of corresponding images each of dimension m.*

*f’(xa , xb) represents the entire set of difference based features computed from the embeddings in space Rm.*

*f’ will represent the final feature set for the generalized linear logistic loss function where the response will be the images are same (0) or blind forged (1)(y)*

*Loss Function: g(sigmoid(Σ(w\*f’+b)),y)*

*where g() is the Logistic Loss Function and y is the response.*

**Step 2: Detecting Skilled forgeries**

1. **Image Feature Extraction**
2. Some of the most important features are the signature image area, the signature height and width, the ratio between the signature length and its width etc. can be computed.
3. Histogram of oriented gradients-based features can be extracted from the images.
4. Fourier transform based features can also be computed for the images.

These features can be used in conjunction to the below methodology.

1. **Deep Triplet Ranking network-based features**

In step 2, we build a more sophisticated model to detect fraud when the signatures look very identical, that is in case of skilled forgery. Here the differences between the images of genuine and forged signatures are very minute and is challenging to detect by even a trained eye.

We use Siamese convolution neural network to train the model here as well, but using *triplets* instead of pairs. This is done by taking an *anchor image* (genuine signature of a person) and comparing it with both a *positive sample* (another genuine signature of the same person) and a *negative sample* (a forged signature by someone else of the same person). The image embeddings are created in such a way that the dissimilarity between the anchor image and positive image must be low and the dissimilarity between the anchor image and the negative image must be high for every triplet. This kind of architecture ensures that even small differences in signatures can be captured in order to flag a skilled forgery as shown in Fig.2.

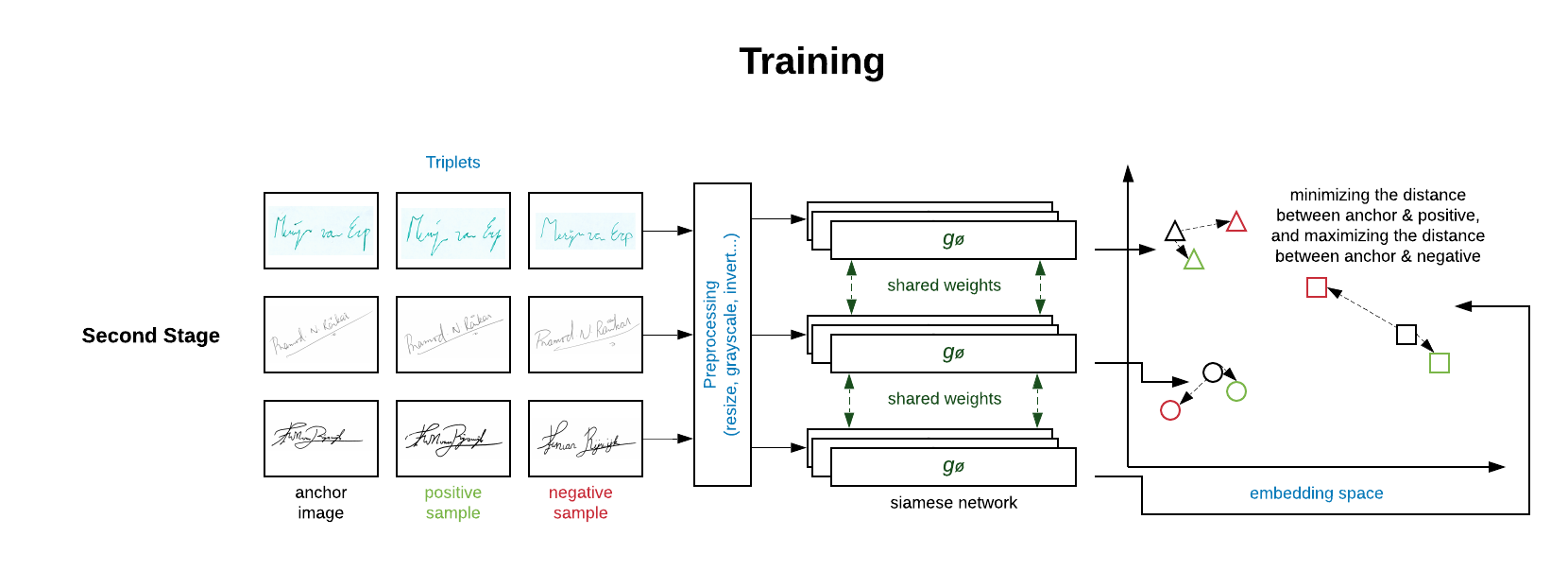
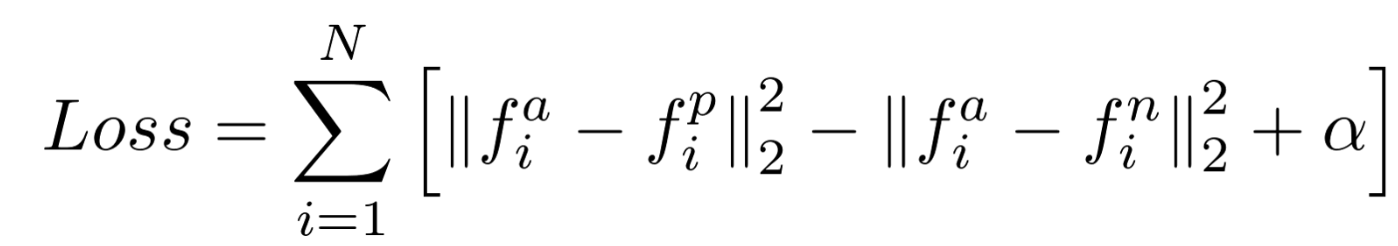


Fig.2: Second stage to detect skilled forgery using Triplet Ranking CNN Network



where,

*f(a) refers to the image encoding of the anchor a*

*f(p) refers to the image encoding of the positive p*

*f(n) refers to the image encoding of the negative n*

*alpha is a constant used to make sure that the network does not try to optimize towards f(a) - f(p) = f(a) - f(n) = 0*

**Bayesian Optimization and Gaussian Process**

The model hyperparameters can be optimized using Bayesian optimization which can save a lot on the time complexity of the grid search with selecting the best hyperparameter at each iteration. One very important aspect is to correctly estimate the value of the *threshold alpha* as in the triplet loss function using Bayesian theory can be applied as shown in Fig.3.

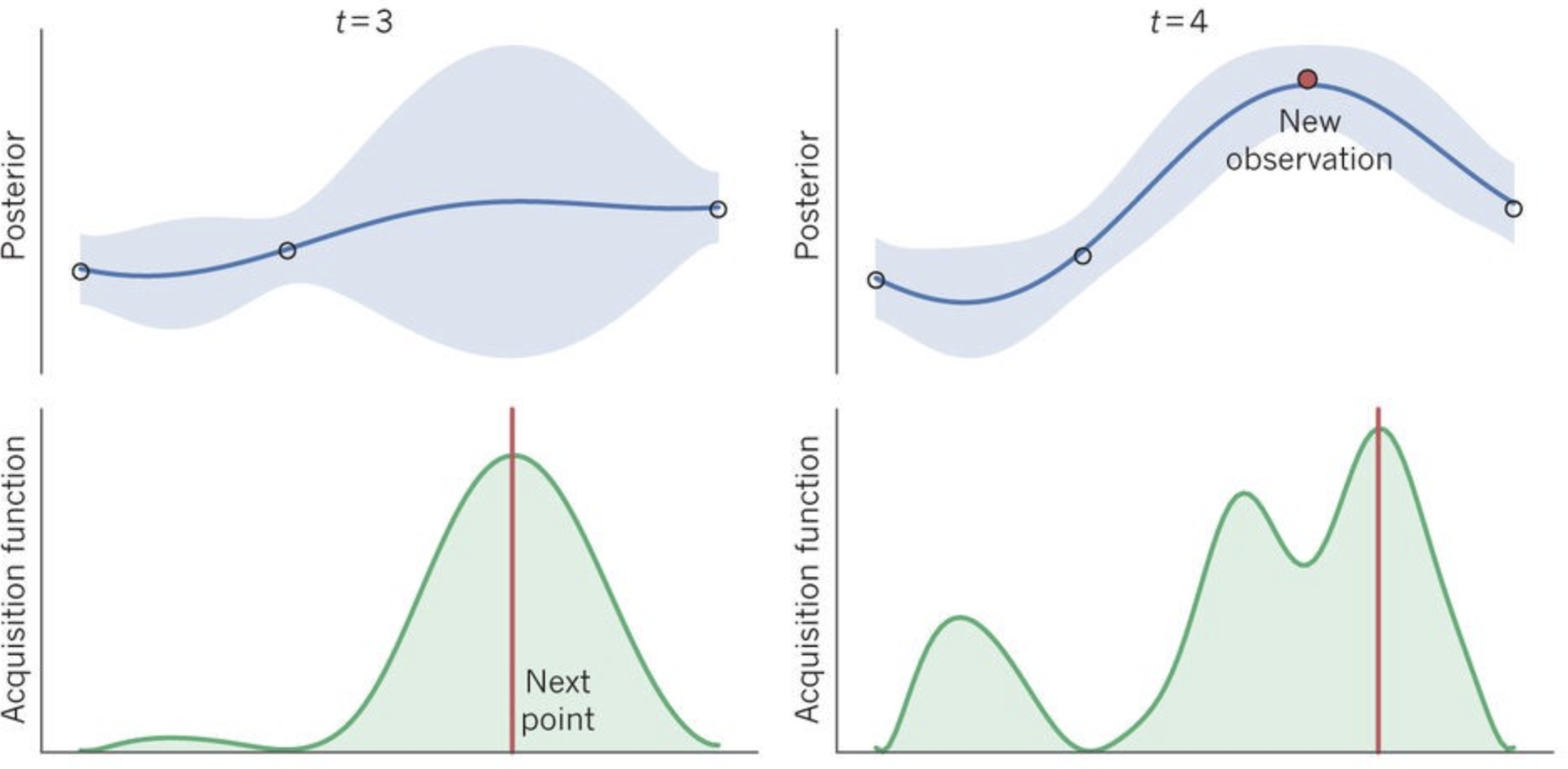


Fig.3: Bayesian Optimization and Gaussian Process

**Final Workflow in Forged Signature Detection**

Once the two-step training process is completed, all we need is the trained model outputs from the two models. Next, we create a database framework to save the original signatures of every new customer that the bank acquires against a unique ID.

The testing process is depicted in Fig.4.

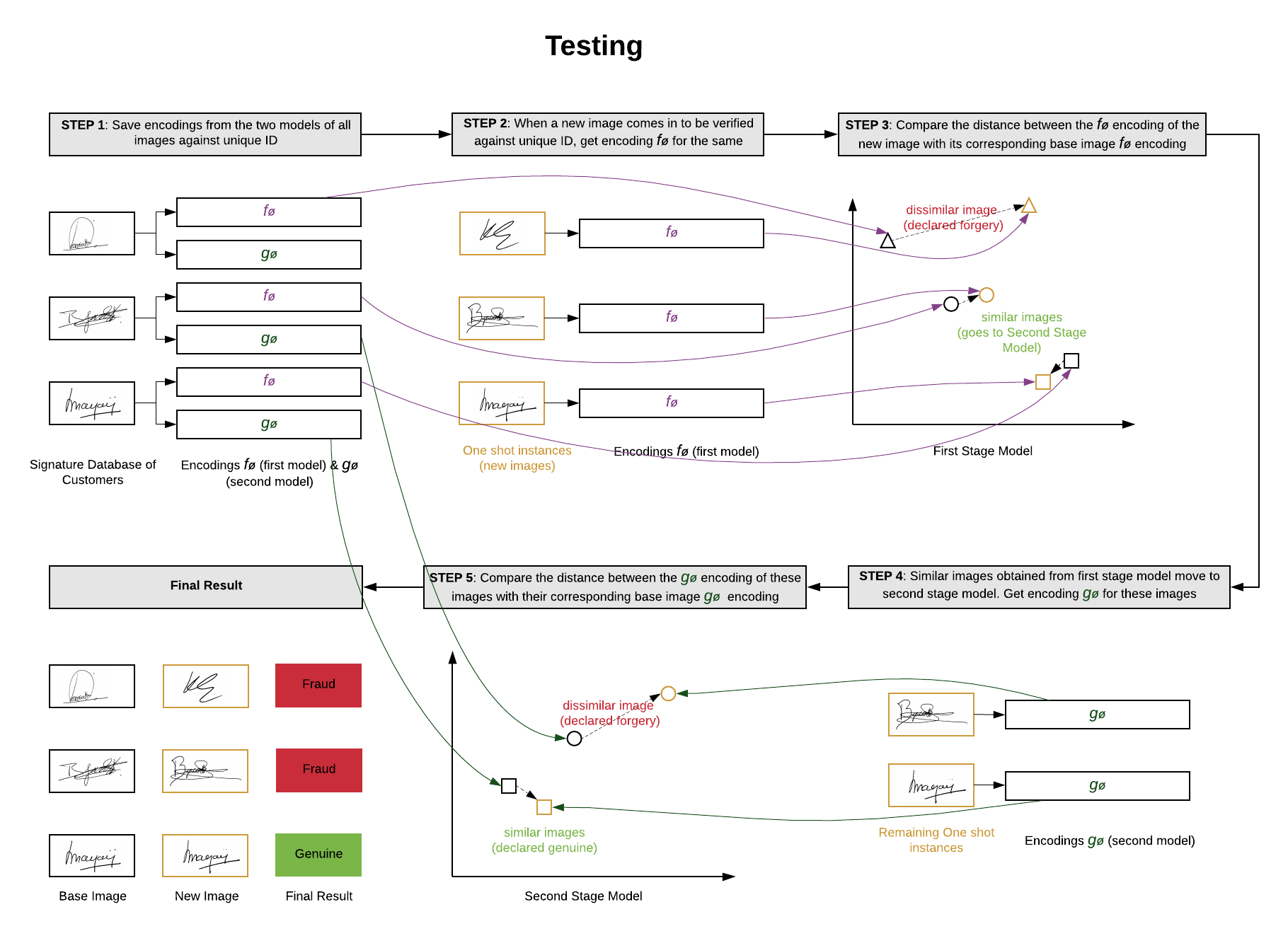


Fig.4: Test Image Forgery Detection using Hierarchical Triplet Ranking Network

We pass these *base* images through the models we obtained in the training process and get the corresponding encodings. This is in the format of a vector of length 128. We precompute these encodings and save them against the individuals’ unique ID in the database. Now, whenever we acquire a new signature image against one of the unique ID’s that needs to be accessed to determine whether it is genuine or fraud, the machine passes that image through the model obtained in step 1 to get its encodings.

This new encoding is then compared to its corresponding encoding of the *base* image of that individual to determine whether it is similar or not. If in this first step it is determined to be *not similar*, then the machine stops there and declares the image to be a fraudulent signature. Conversely, if it is found to be similar, then the machine goes to step 2 and gets the encoding from the second model and compares it to the corresponding base image encoding of second model. If the distance between even these encodings is found to be close, i.e., similar images, then the machine declares the new image as a genuine, otherwise it is considered a forgery.

The entire workflow is shown below in Fig.5.

**Hierarchical Siamese Triple Ranking Convolution Network in Signature Forgery Detection - Entire Workflow Diagram**

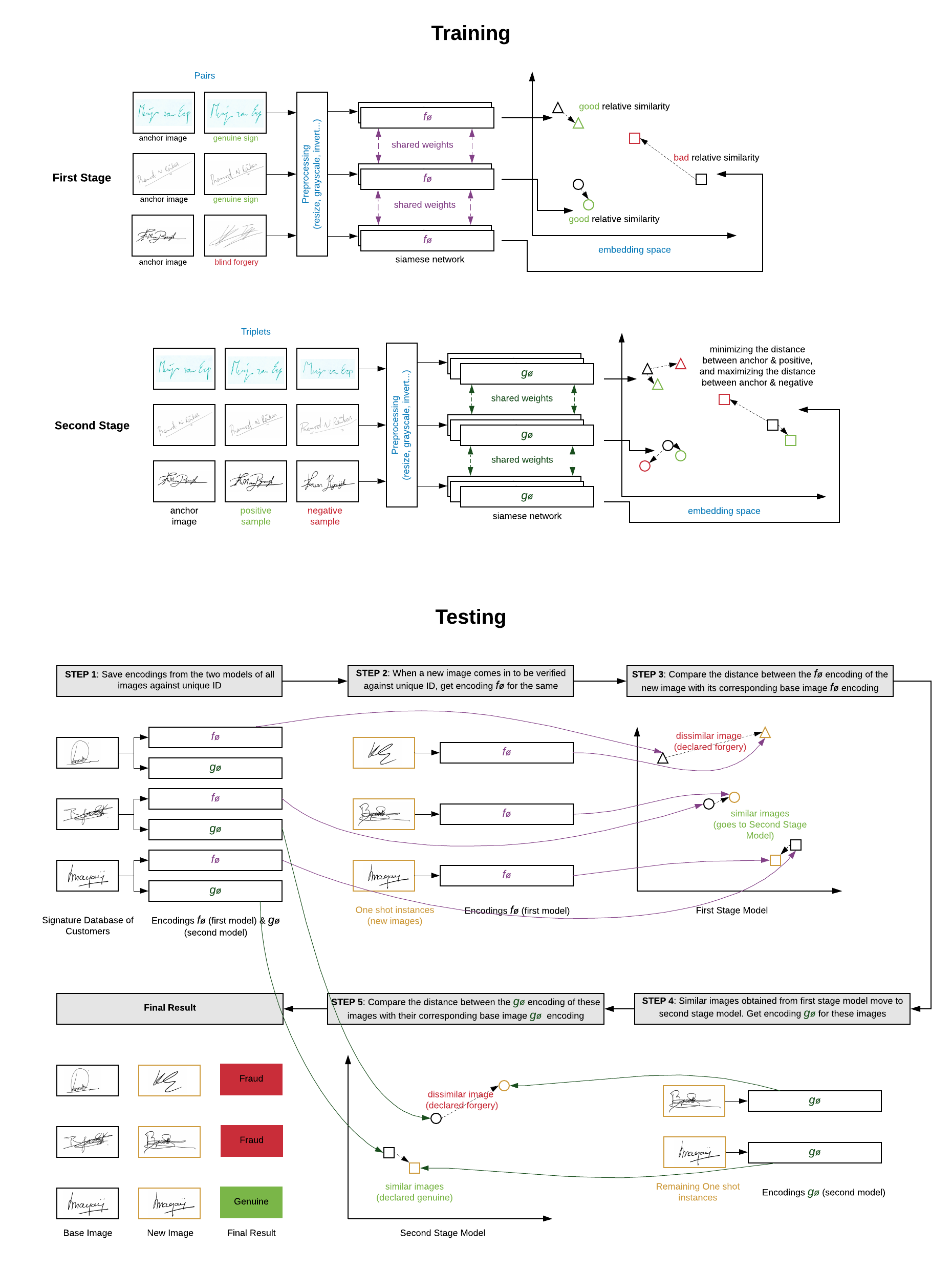


Fig.5: Entire Workflow Diagram for Signature Forgery Detection