Have we met before: ID'ing face

Presented By:
Nikhila, Sasikanth and Adhun

Presentation Sub-sections

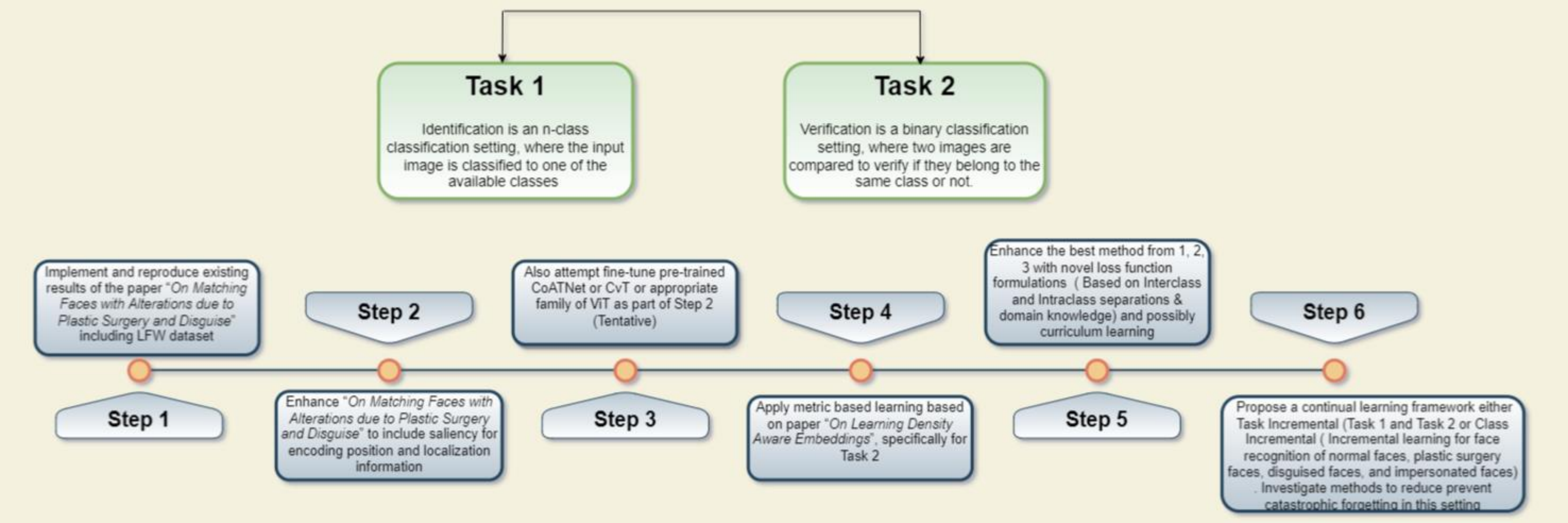
- Introduction
- Our Proposals
- Dataset details
- Experiments carried out
- Results
- Conclusion

Introduction

- Many techniques in the fields of machine object recognition and pattern recognition rely heavily on face detection.
- Face domain can be categorized mainly into two – Identification and Verification.
- Automated Face recognition has been one of the most revolutionary breakthroughs in the last decade. Especially with the projects as India's Aadhar and Apple's Face ID, which opened the door to a plethora of new research opportunities.
- With the advancements, there were many challenges especially in variations like in plastic surgery and disguise.

- Convolutional Neural Networks (CNN) as well as other classification models use a deep metric learning-based loss function to learn discriminative embeddings.
- The loss function tries to bring the embeddings of the same classes in the output manifold closer together.
- A simple computation of the distance in this embedding space yields the dissimilarity score between the two pictures.
- There are many applications making use of deep metric learning algorithms - person reidentification, 3D object retrieval, biometric recognition, robot perception, patch matching and object recognition.

Our Proposals



We have carried out the following as part of our project:

Task1: Implemented and reproduced existing results of the paper "On Matching Faces with Alterations due to Plastic Surgery and Disguise" including LFW dataset.

Task2: Reproduced the results of the paper "Escaping the Big Data Paradigm with Compact Transformers".

Task3: Implemented and reproduced existing results of the paper "On Learning Density Aware Embeddings".

Task4: Proposed the novel architecture – LightAttentionCNN29.

Task5: We have carried out various experiments using different loss functions for different models. We have either used pretrained models or fine-tuned them for the experiments.

Dataset details

- DFW2018 Dataset - Protocol1 (Impersonation): Genuine set contains pairs having '1', and imposter set contains pairs with '3'.
  - Number of samples in train : 4997
  - Number of samples in test : 25046
  - The mask matrices contain values belonging to {0,1,2,3,4}. Here,
    - 0 - No use
    - 1 - Genuine Validation
    - 2 - Genuine Disguise
    - 3 - Imposter Impersonator
    - 4 - Cross-subject Imposter

- That is, the value of element (i,j) specifies whether the pair created by the ith and the jth image is of no use, a genuine validation pair, a genuine disguise pair, an imposter impersonator pair, or an impostor cross-subject pair.
- The ordering of images is the same as provided with the dataset in the "Training_data_face_name.txt" and "Testing_data_face_name.txt“ text files. training_data_mask_matrix.txt contains a 3386x3386 matrix and testing_data_mask_matrix.txt contains a 7771x7771 matrix.
- Note that both these matrices are symmetric matrices since pairs (i,j) and (j,i) refer to the same pair.
- The training and testing mask matrices can be used for extracting the relevant pairs/scores for the three protocols as follows:
    - Protocol-1 (Impersonation): Genuine set contains pairs having '1', and imposter set contains pairs with '3'.

- LFW Dataset
  - LFW dataset has 13233 images, 5749 people and 1680 identities with two or more images.
  - Dataset split into Train and Test. The train has 1180 identities and test has 500 identities.
  - Some identities have just single face image.
  - Test contains Probe & Gallery with non-overlapping face images.
  - Train dataset , again split (stratify split) into Train and Validation in the ratio of 70-30 during fine-tuning.

- Plastic Surgery Face Database
  - Number of samples in train : 648
  - Number of samples in validation : 72
  - Number of samples in test : 180

Experiments carried out

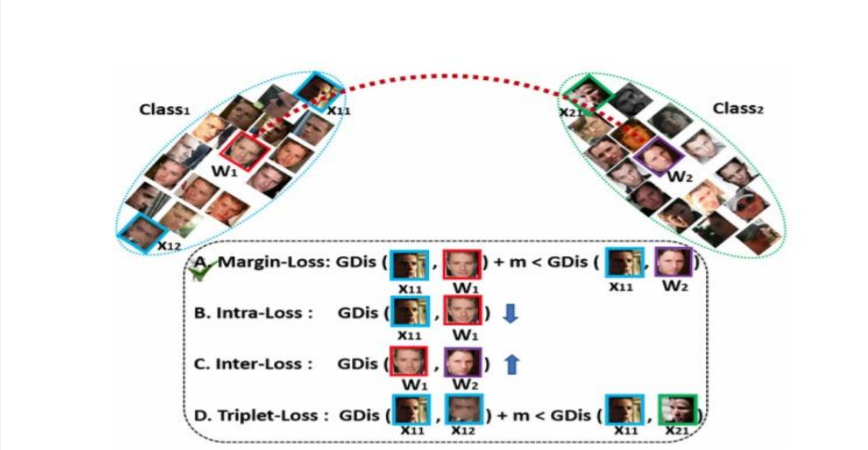
- **Using LFW dataset:**
  - We used this dataset for face identification tasks.
  - Models used:
    - LightCNN29 pretrained init (pretrained and fine tuned with different loss functions combinations)
    - Compact Convolution Transformer (pretrained and fine tuned with different loss functions combinations)
  - Loss functions used:
    - ArcFace
    - Contrastive + ArcFace
    - Contrastive + Variance
    - SemiHard Triplet
    - SupCon with SNR Distance

- **Using Plastic Surgery Face Database:**
  - We used this dataset for face verification tasks.
  - Models used:
    - LightCNN29 (fine tuned)
  - Loss functions used:
    - Contrastive

- **Using DFW dataset:**
  - We used this dataset for face verification tasks.
  - Models used:
    - LightCNN29 (fine tuned)
    - LAttCNN29 (Light Attention CNN 29)
  - Loss functions used:
    - Contrastive
    - Center + Angular
    - Cosine Augmented Center loss
    - ArcFace + Variance Loss

Loss Functions Used

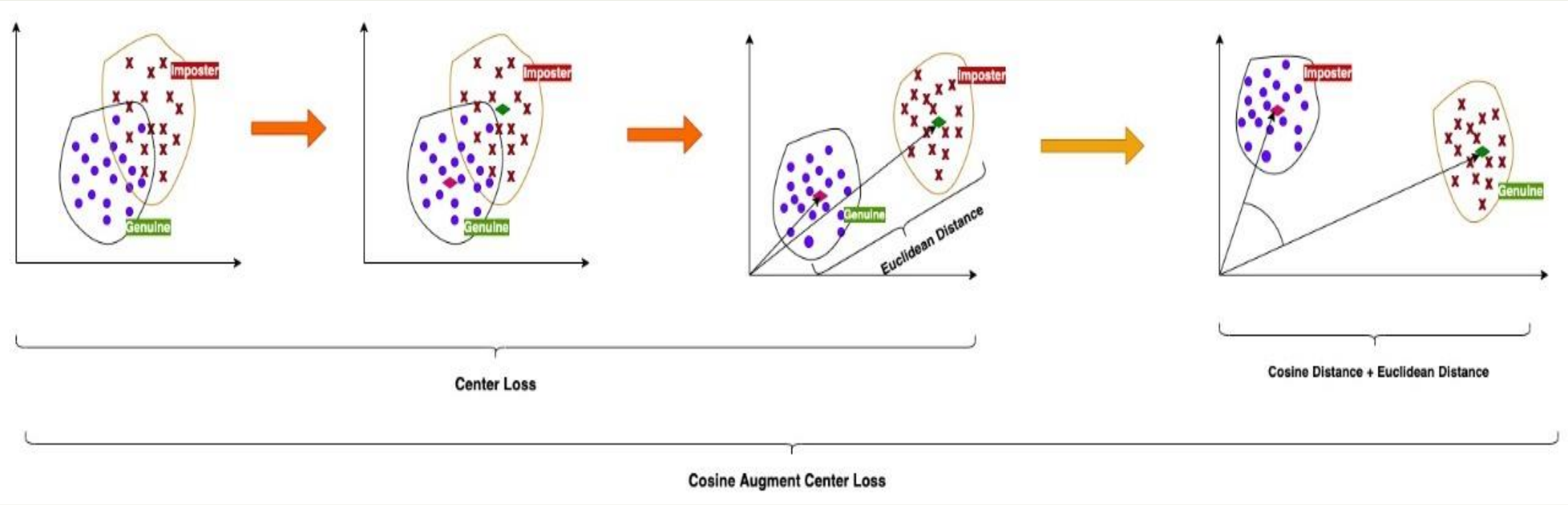
- **Additive Angular Margin Loss (ArcFace):** ArcFace has a clear geometric interpretation due to the exact correspondence to the geodesic distance on the hypersphere



- **SupCon - Supervised Contrastive Loss :** extended self-supervised batch contrastive approach to the fully-supervised setting, allowing us to effectively leverage label information. Clusters of points belonging to the same class are pulled together in embedding space, while simultaneously pushing apart clusters of samples from different classes.

Proposed Loss Function for DFW

- **Center Loss:** Increases the intra-class distance and decreases the inter-class distance
- **Cosine Augmented Center Loss**: Optimizes the cosine distance, in addition to Euclidean distance to further increase inter-class distance



- **Contrastive Loss:** Takes L2 distance between pairs. It maximize the distance between negative pairs and minimize the distance between positive pairs.

- **SNRDistance:** Based on Signal-to-Noise Ratio(SNR) for measuring the similarity of image pairs for deep metric learning

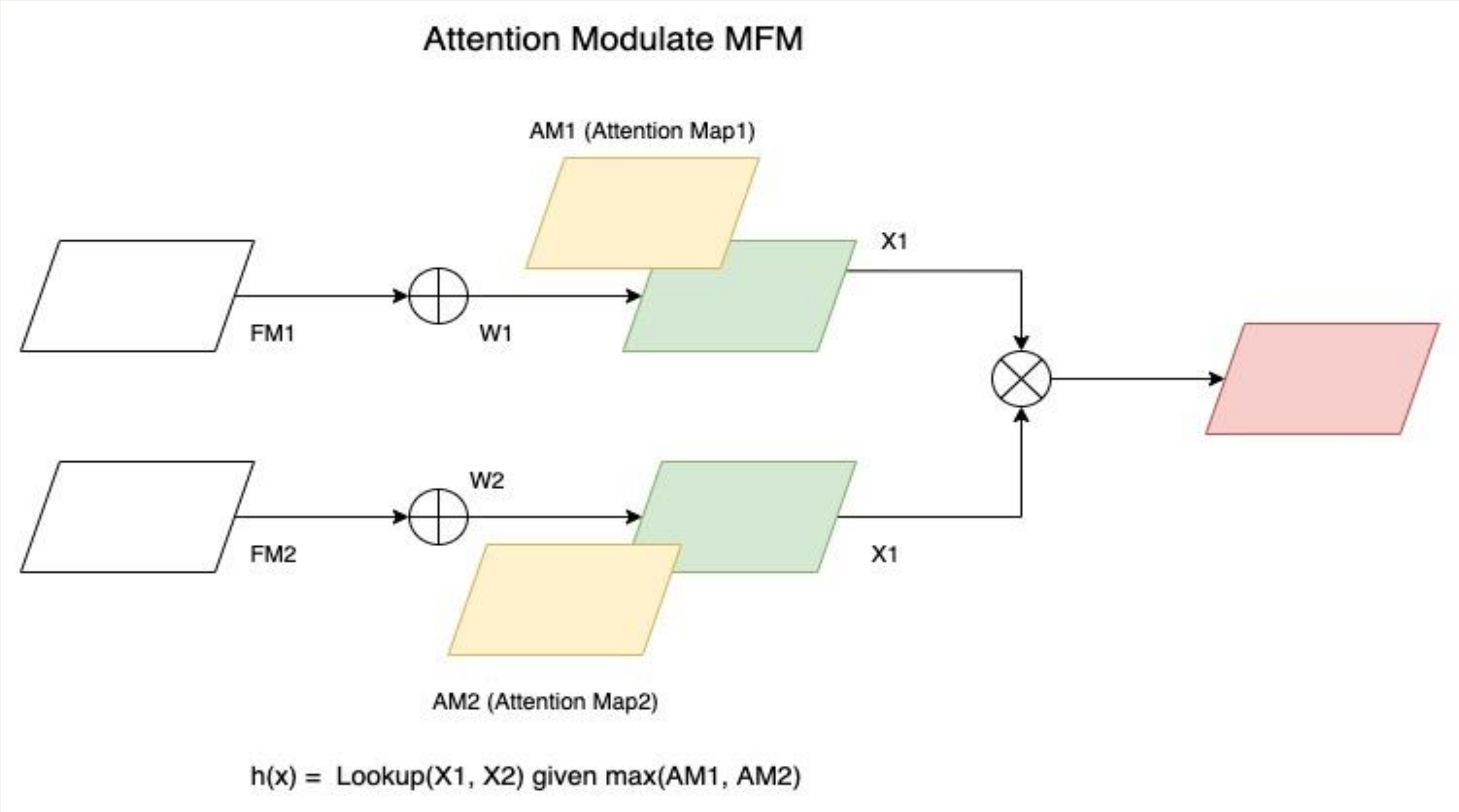
- **Center + Angular:** Maximizes the distance and angle between cluster centers.

- **Cosine Augmented Center loss :** Maximizes the angle between the centers of the clusters.

Attention Modulated MFM

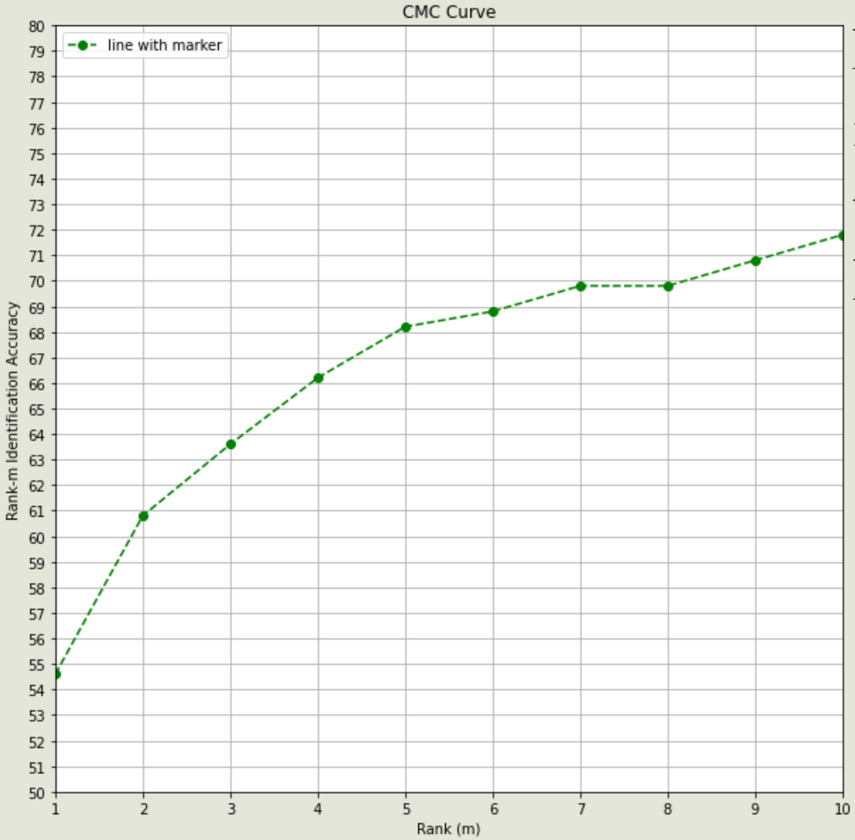
- Obtain attention map using SimAM: A Simple, Parameter-Free Attention Module.
- Obtain MFM based on the max attention values of the corresponding channel attention maps

h(x) = Lookup(x1,x2) given max(AM1, AM2)



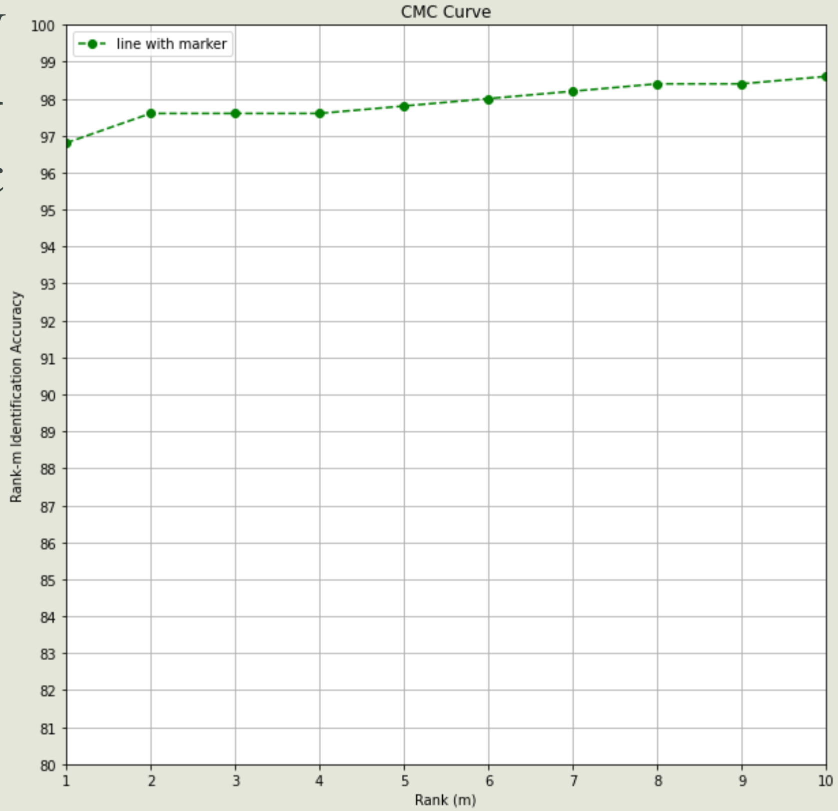
Results

- **Using LFW dataset:** Pretrained models
  - **ArcFace** (LightCNN29)

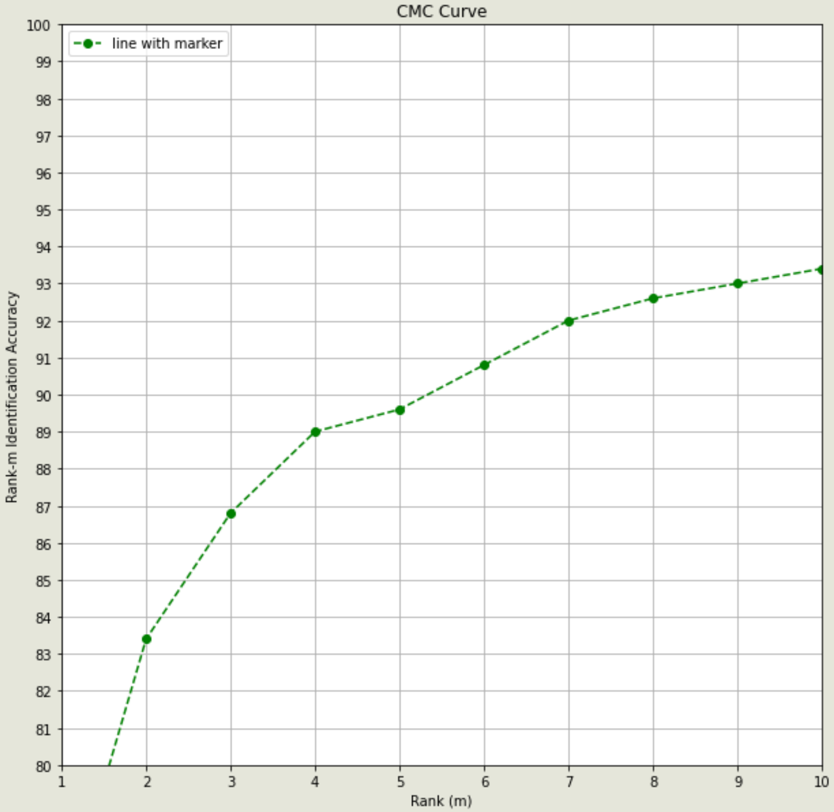


Rank1 Identification accuracy for LightCNN29 using Pre-trained model with only Arc Face Loss(%) 54.6

Rank1 Identification accuracy for LightCNN29 using Pre-trained model before training with ArcFace Losses(%) 96.8

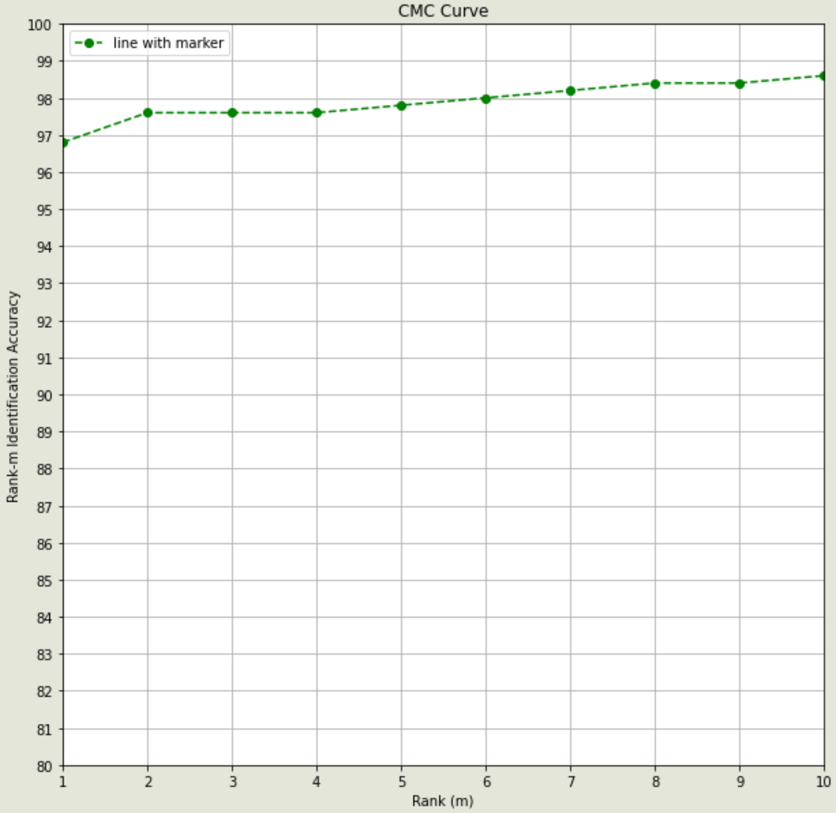


○ **Contrastive + ArcFace** (LightCNN29)

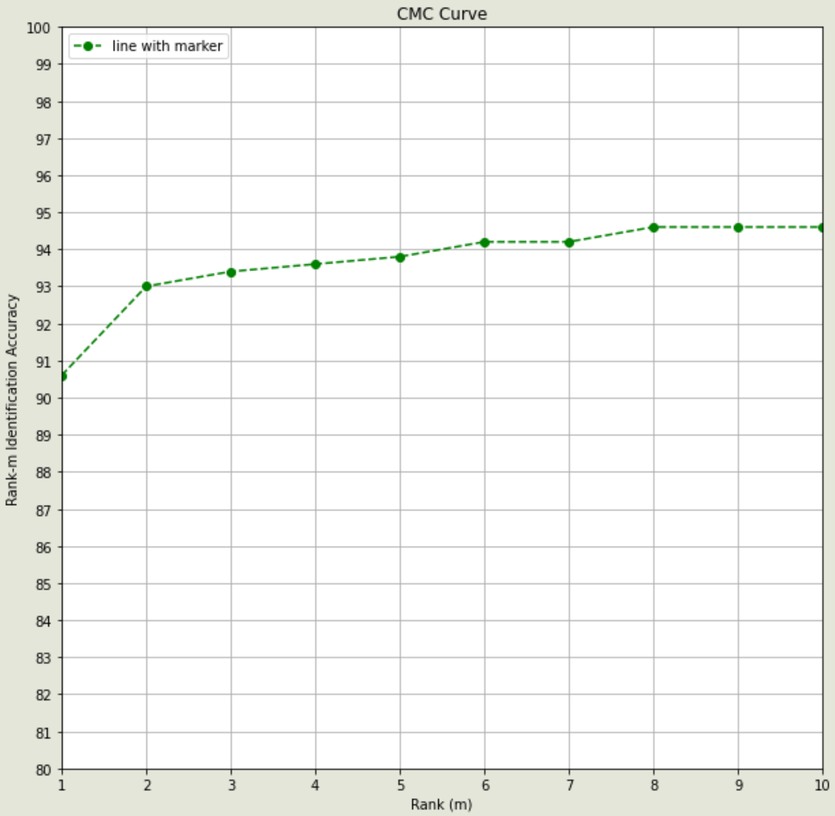


Rank1 Identification accuracy for LightCNN29 using Pre-trained model with Contrastive & Arc Face Losses(%) 75.6

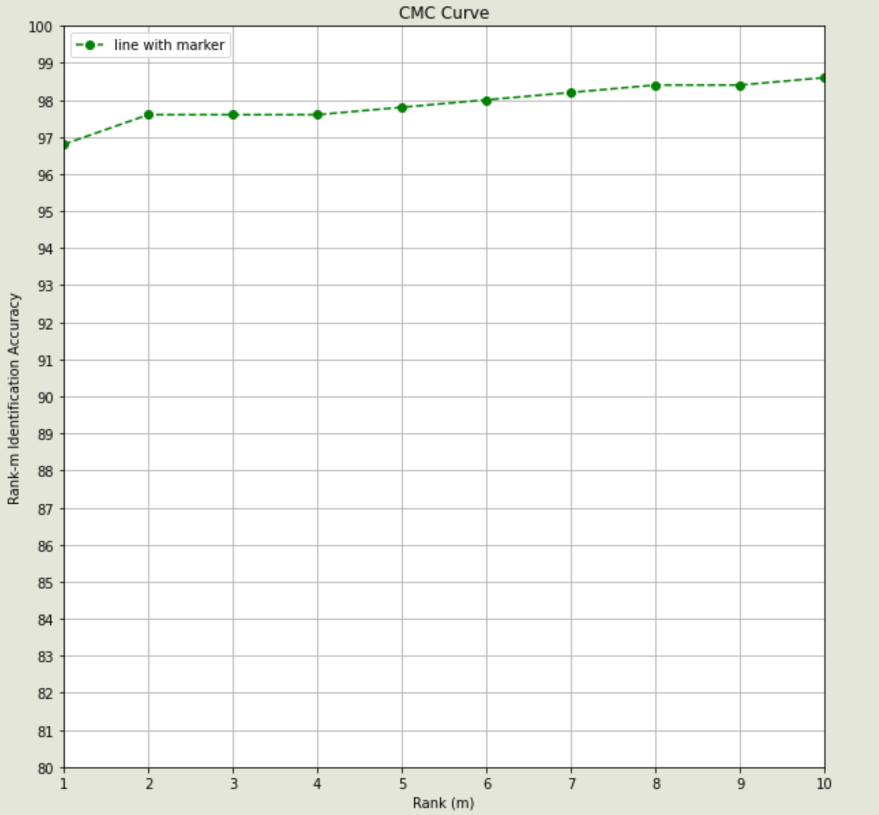
Rank1 Identification accuracy for LightCNN29 using Pre-trained model before training with Contrastive & ArcFace Losses(%) 96.8



○ **Contrastive + Variance** (LightCNN29)

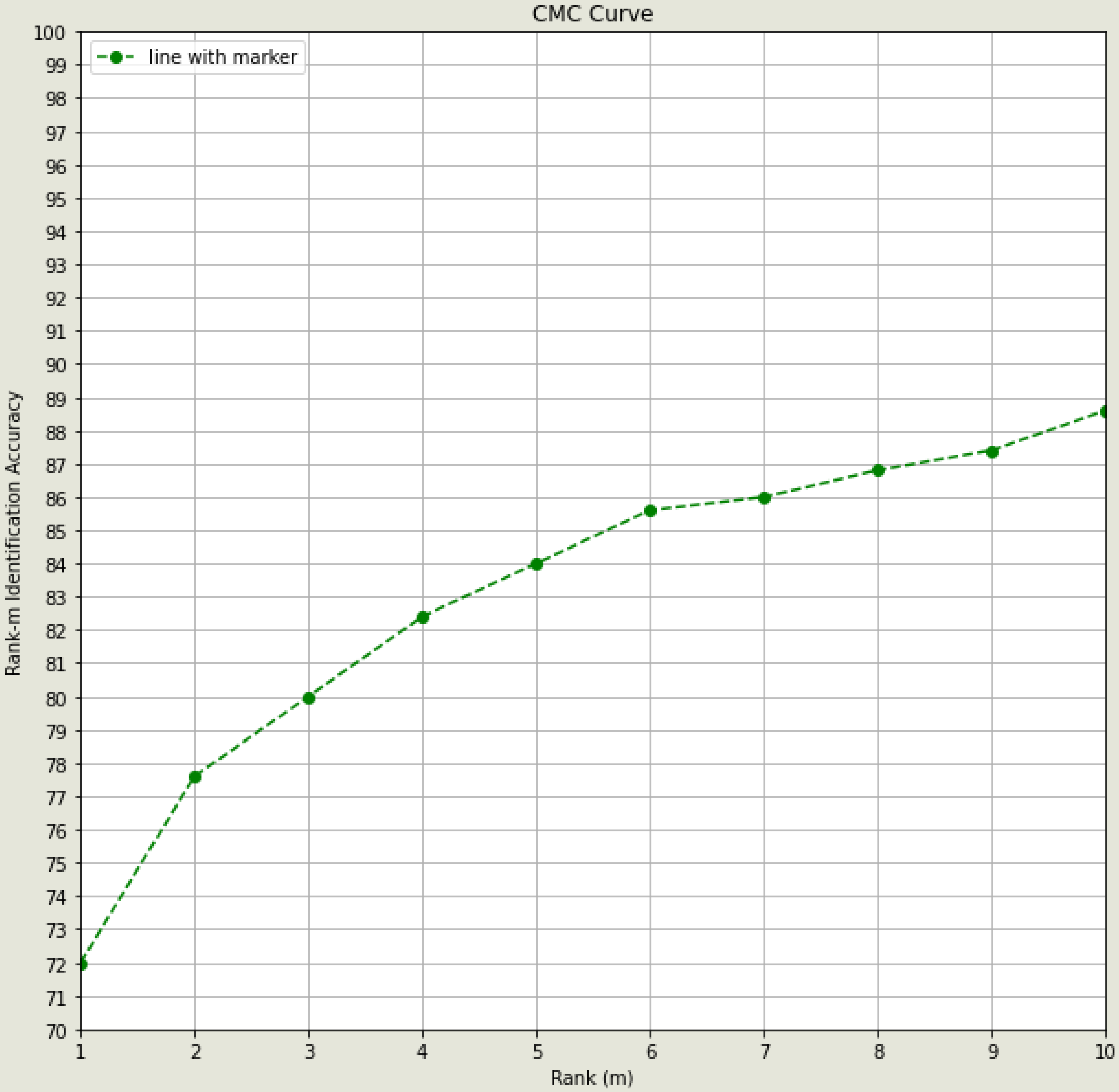


Rank1 Identification accuracy for LightCNN29 using Pre-trained model with Contrastive & Variance Losses(%) 90.60000000000001



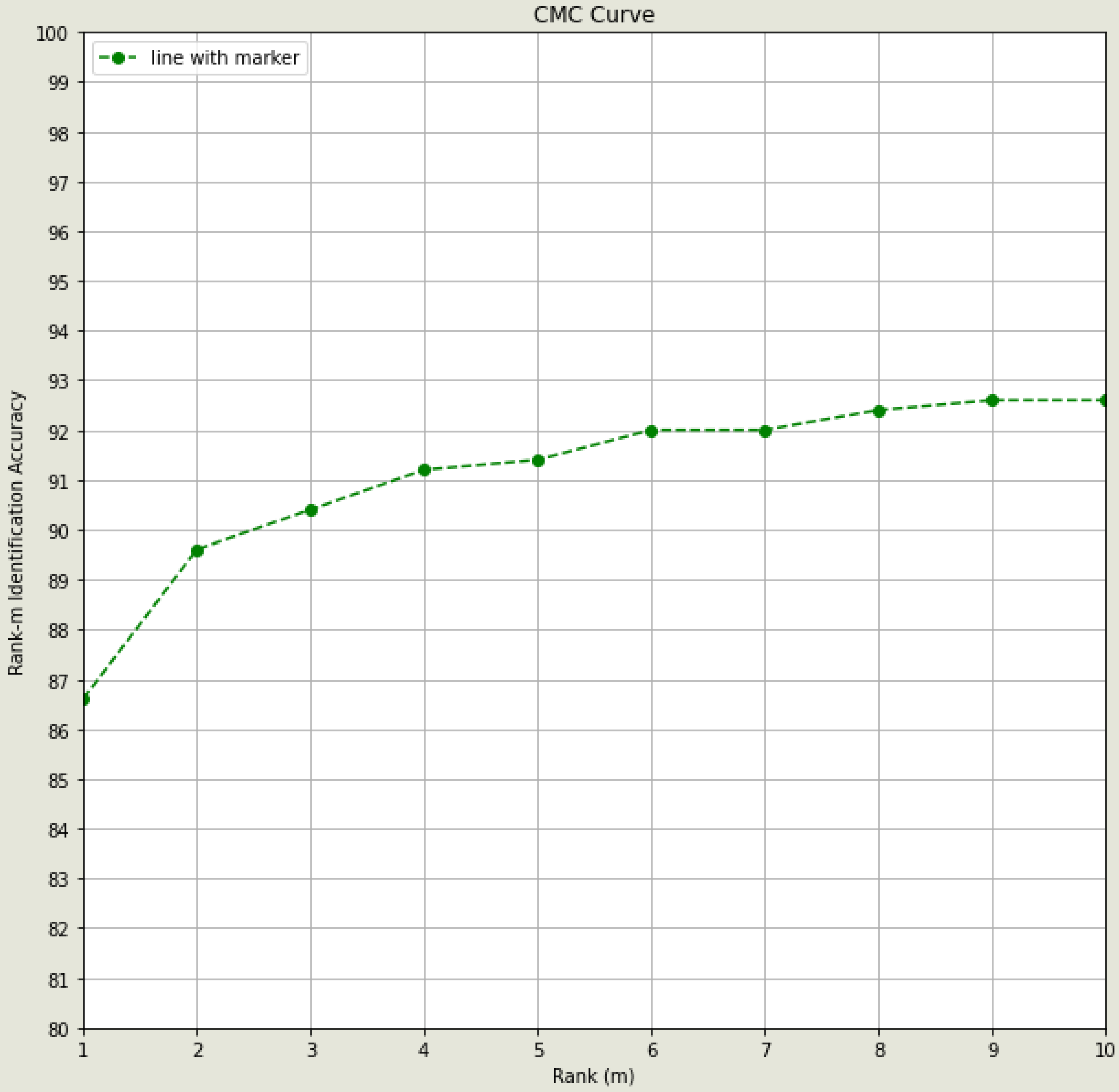
Rank1 Identification accuracy for LightCNN29 using Pre-trained model before training with Contrastive & Variance Losses(%) 96.8

○ **SemiHard Triplet** (LightCNN29)



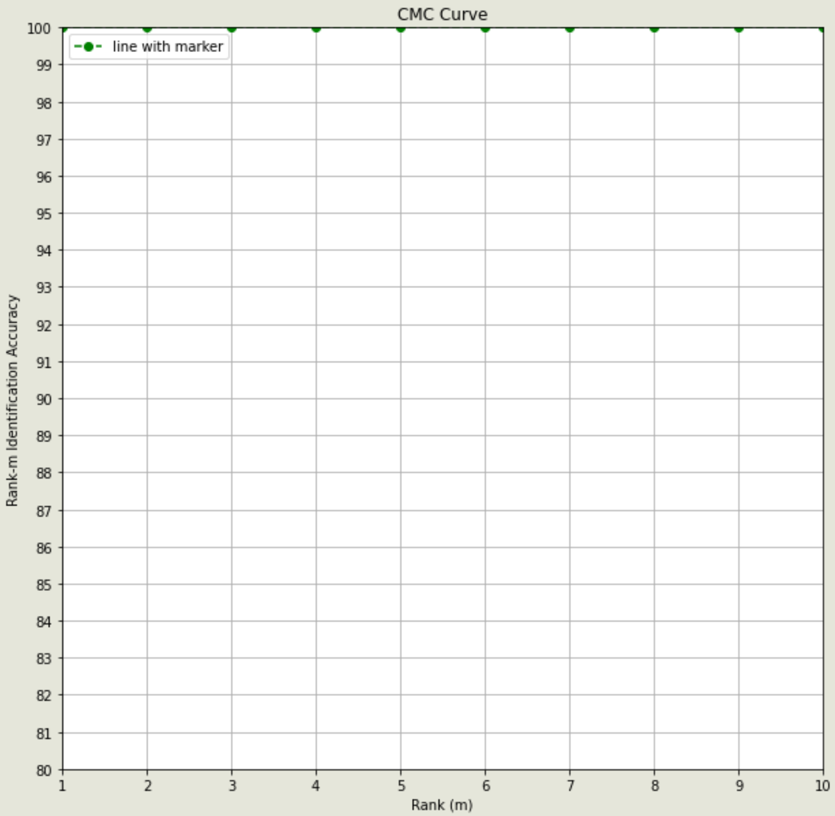
Rank1 Identification accuracy for LightCNN29 using Pre-trained model with Semi Hard Triplet Loss(%) 72.0

○ **SupCon with SNR Dist** (LightCNN29)



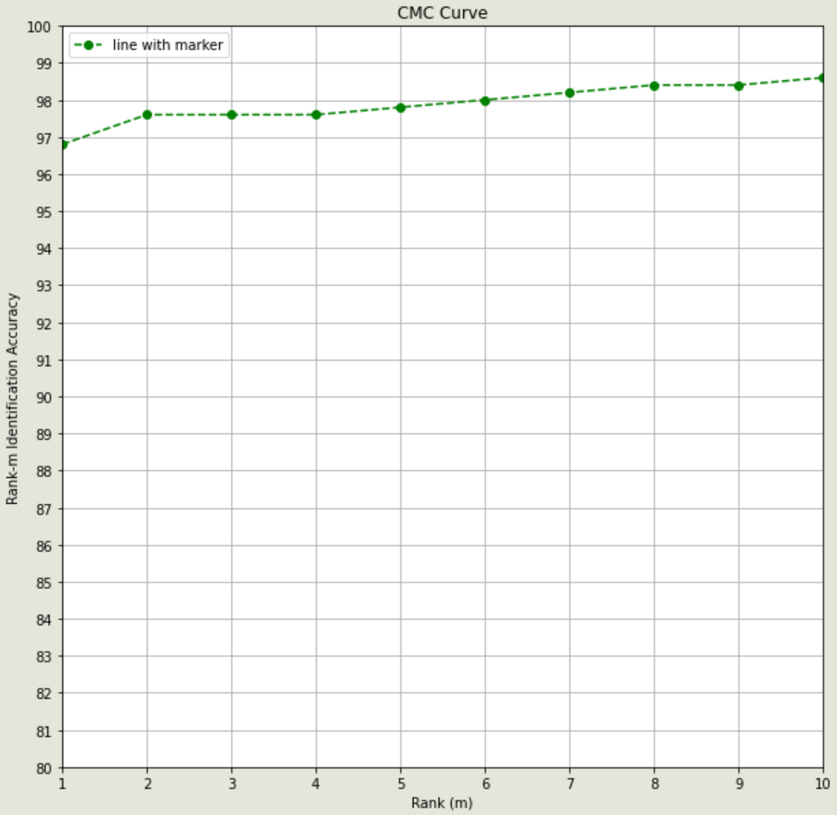
Rank1 Identification accuracy for LightCNN29 using Pre-trained model with Sup Contrastive Loss with SNR distance(%) 86.6

- **Contrastive + Variance** (Compact Convolution Transformer)

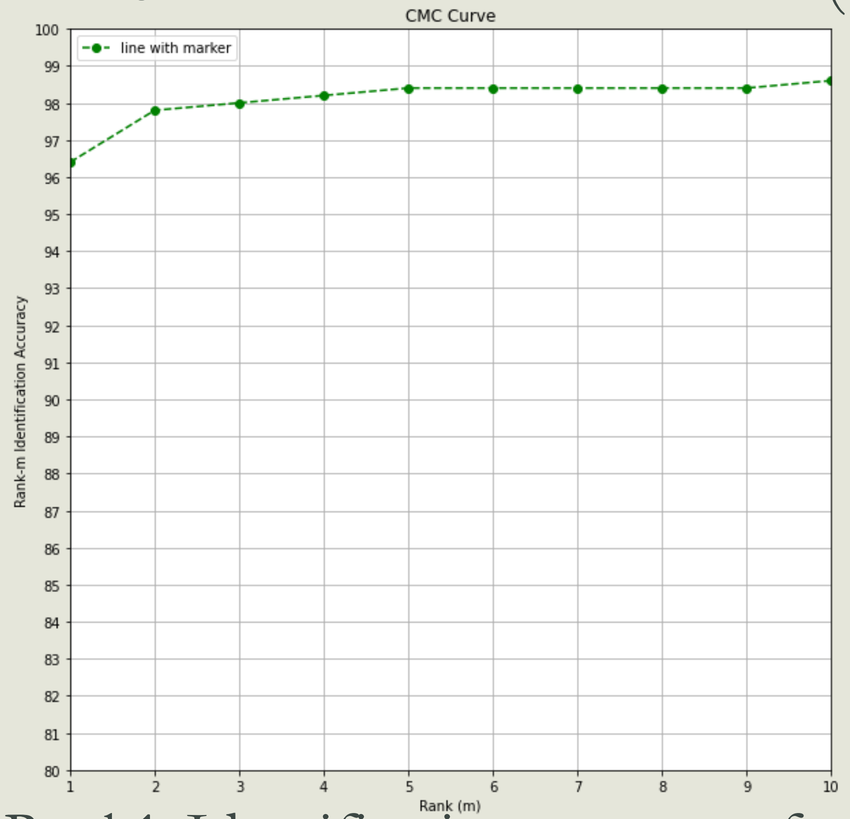


Rank1 Identification accuracy for CCT using Pre-trained model with Contrastive & Variance Losses(%) 100.0

Rank1 Identification accuracy for CCT using Pre-trained model before training with Contrastive & Variance Losses(%) 10.8

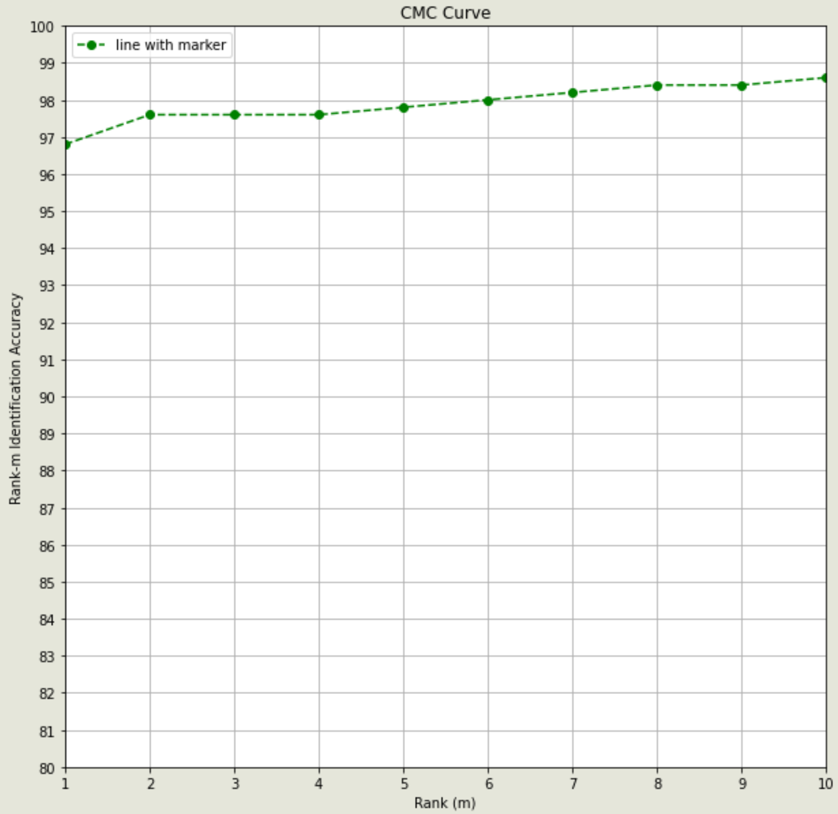


- **Using LFW dataset:** Fine-tuned models
  - **Contrastive + Variance** (LightCNN29)



Rank1 Identification accuracy for LightCNN29 using Pre-trained model after fine tuning with Contrastive & Variance Losses(%) 96.39999999999999

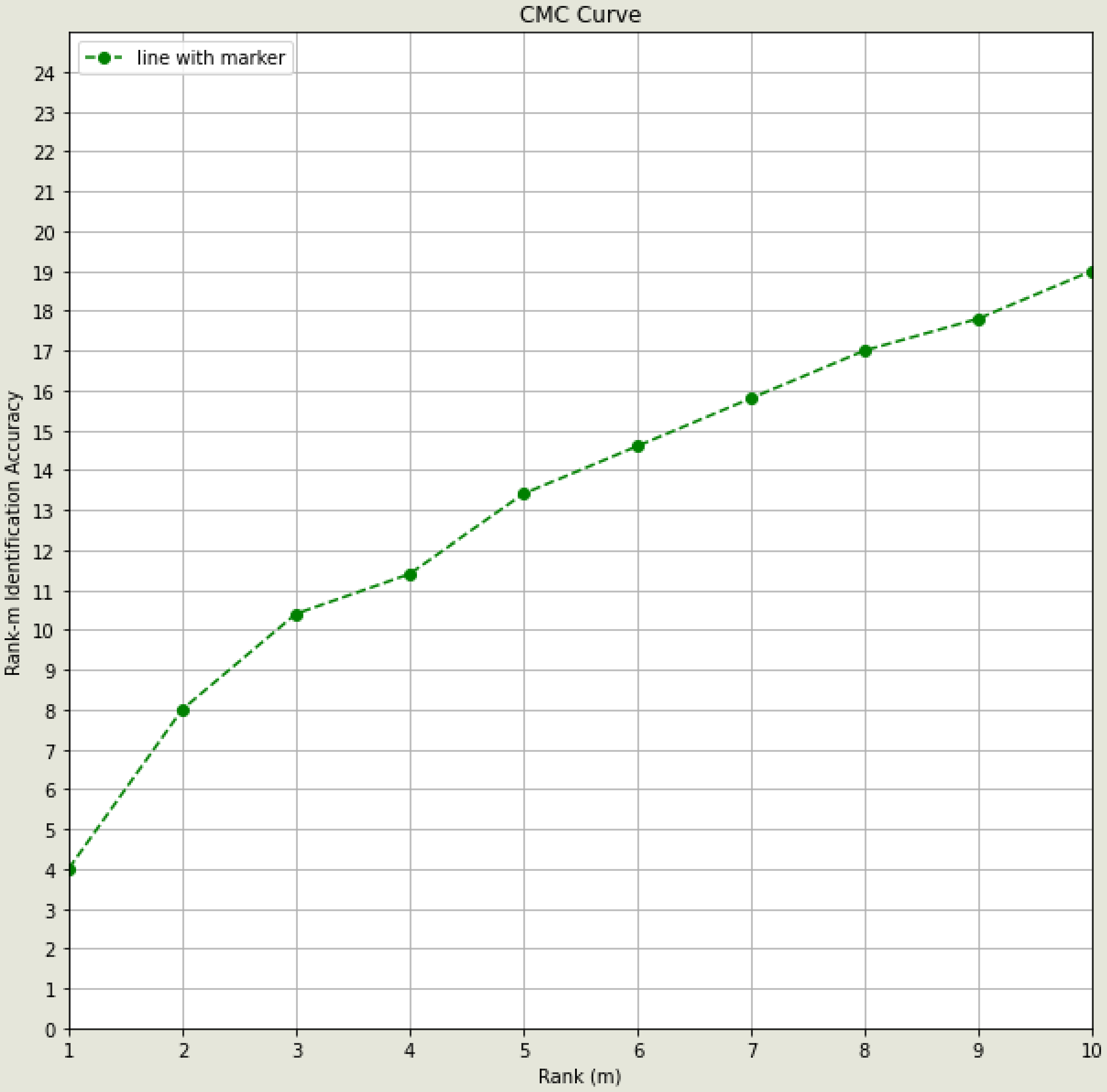
Rank10 Identification accuracy for LightCNN29 using Pre-trained model after fine tuning with Contrastive & Variance Losses(%) 98.6



Rank1 Identification accuracy for LightCNN29 using Pre-trained model before training with with Contrastive & Variance Losses(%) 96.8

Rank10 Identification accuracy for LightCNN29 using Pre-trained model before training with with Contrastive & Variance Losses(%) 98.6

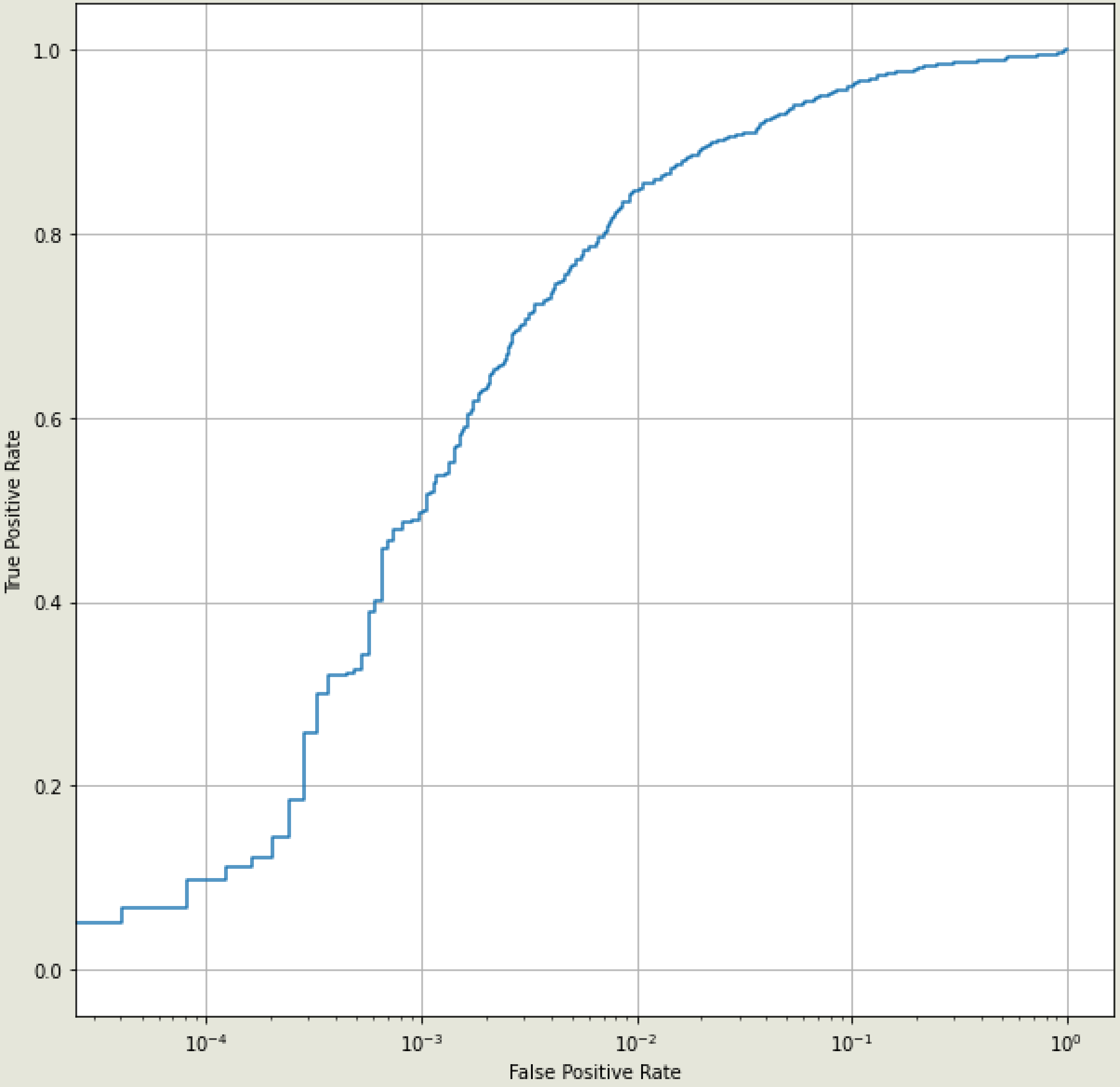
○ **Contrastive + Variance** (Compact Convolution Transformer)

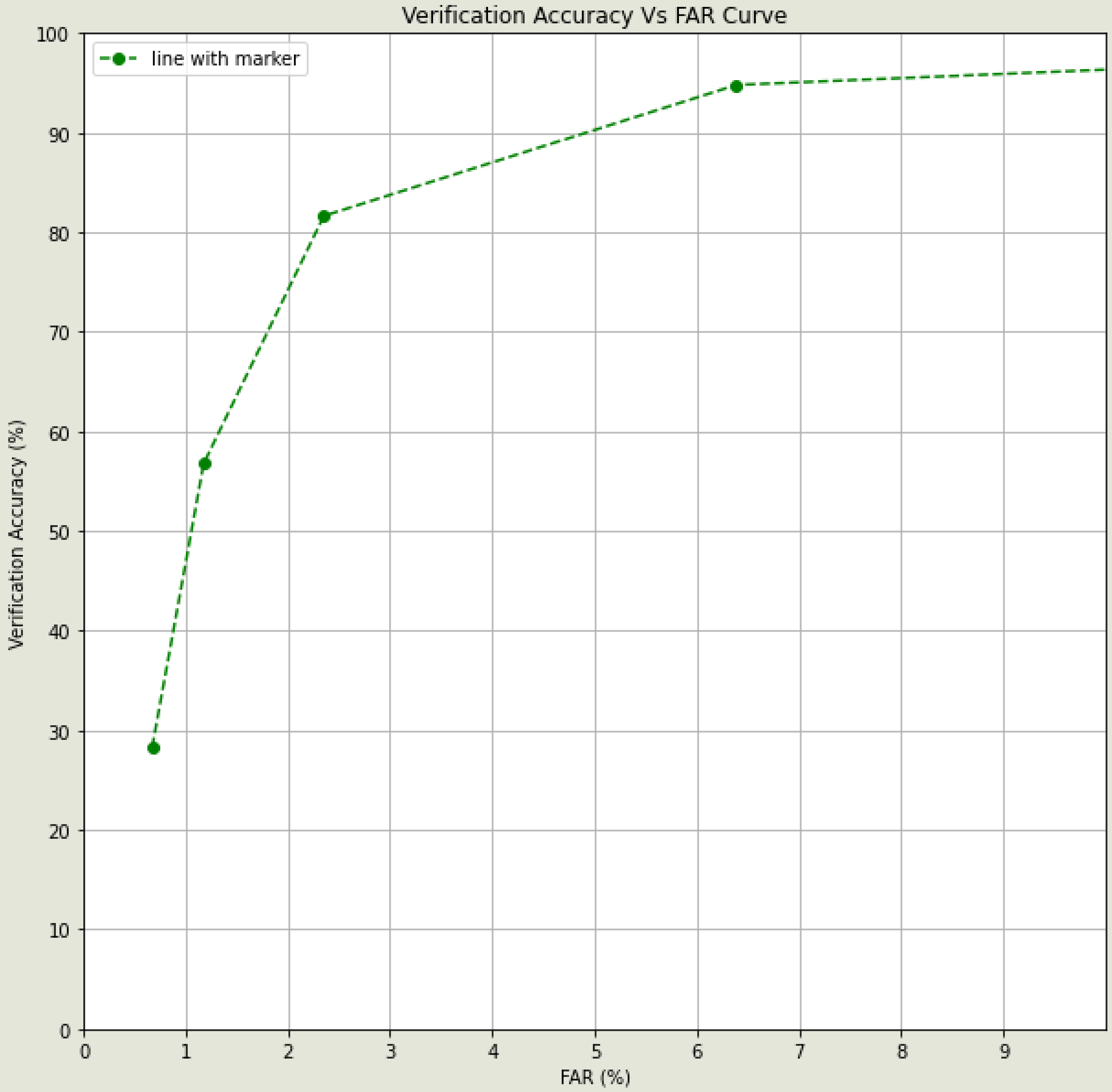


Rank1 Identification accuracy for CCT using Pre-trained model after fine-tuning with Contrastive & Variance Losses(%) 4.0

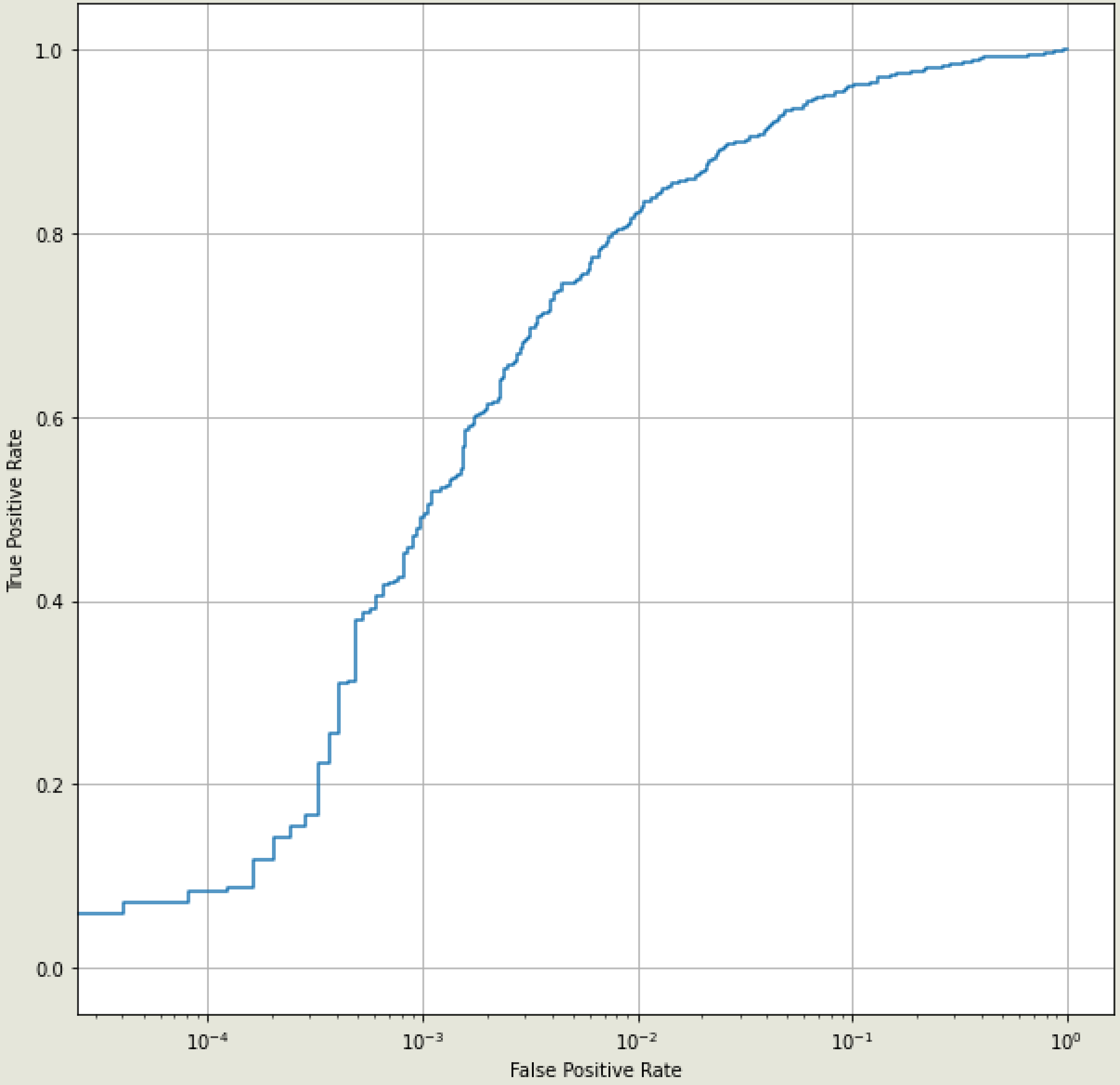
Rank10 Identification accuracy for CCT using Pre-trained model after fine-tuning with Contrastive & Variance Losses(%) 20.0

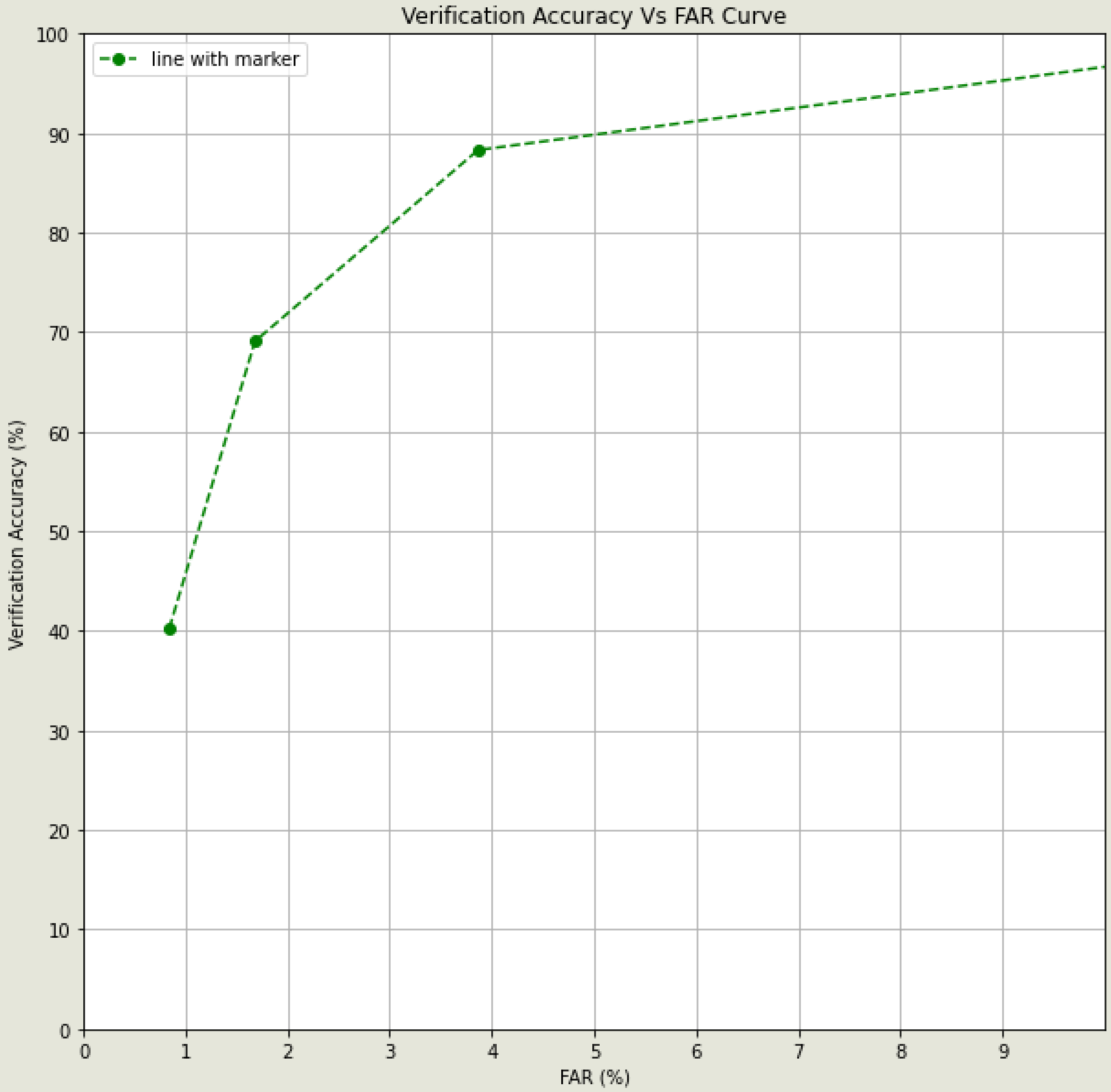
- **Using DFW dataset:** Fine-tuned models
  - **Contrastive** (LightCNN29)



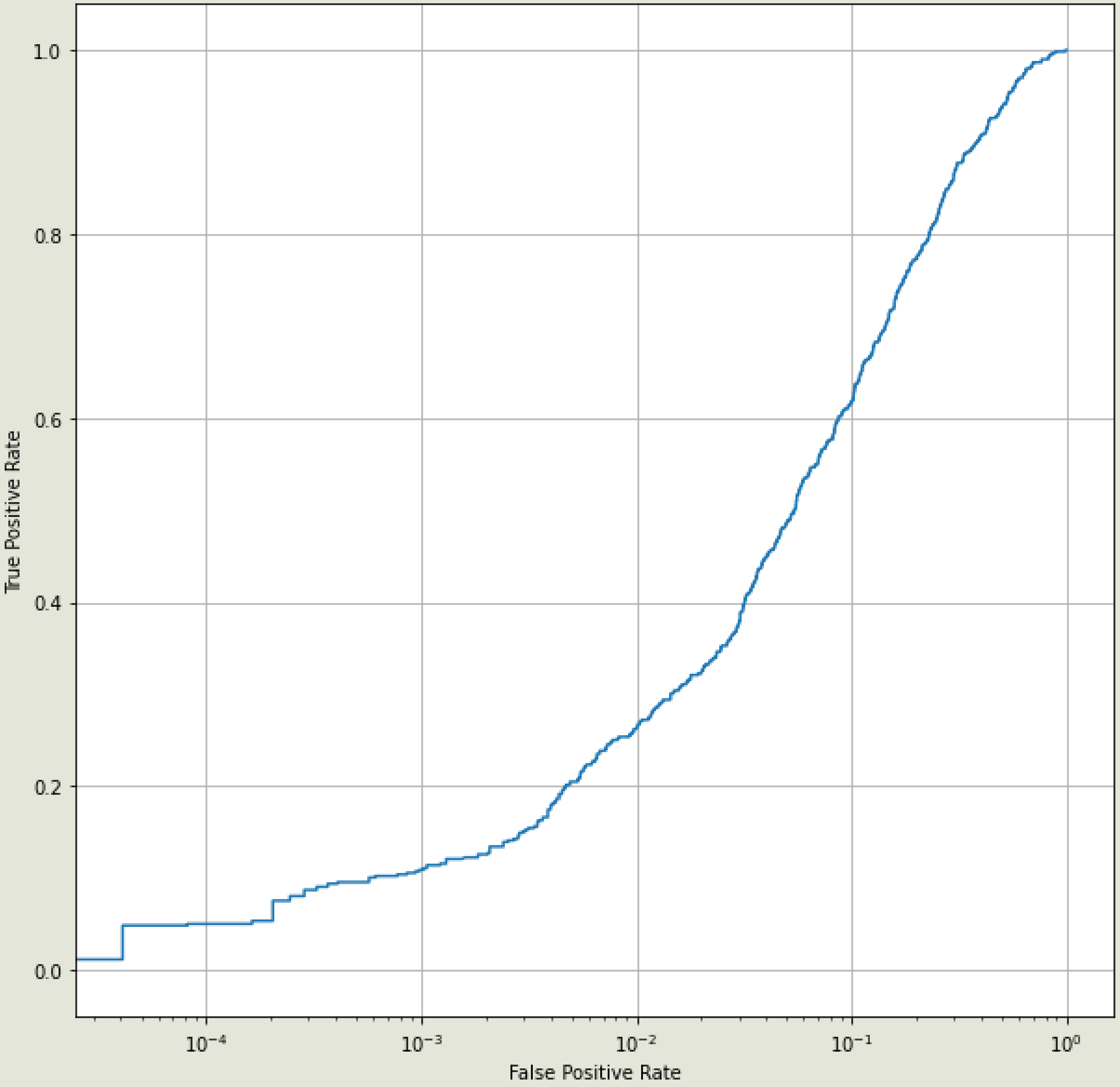


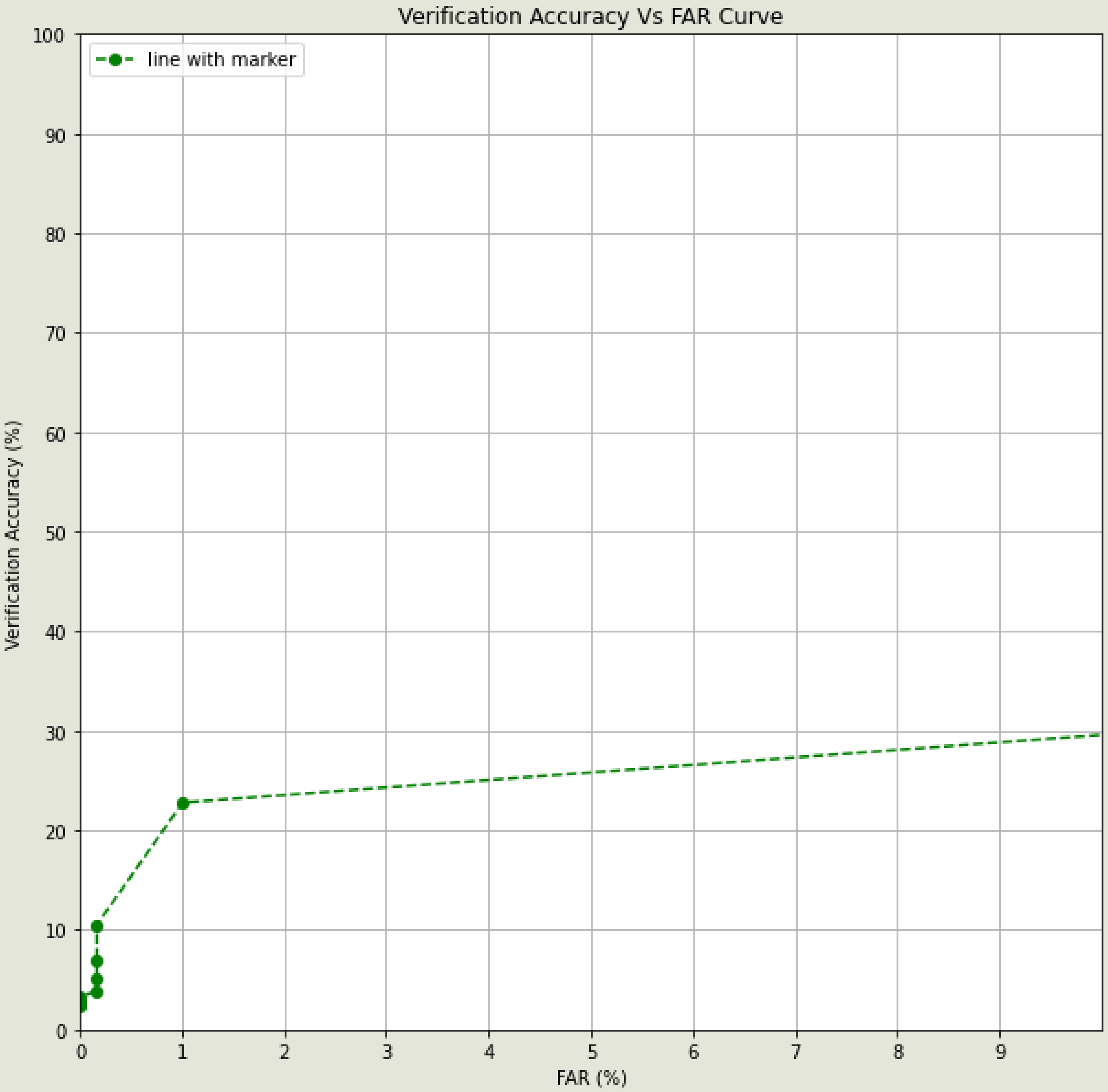
○ **Center + Angular** (LAttCNN29 (Light Attention CNN 29))



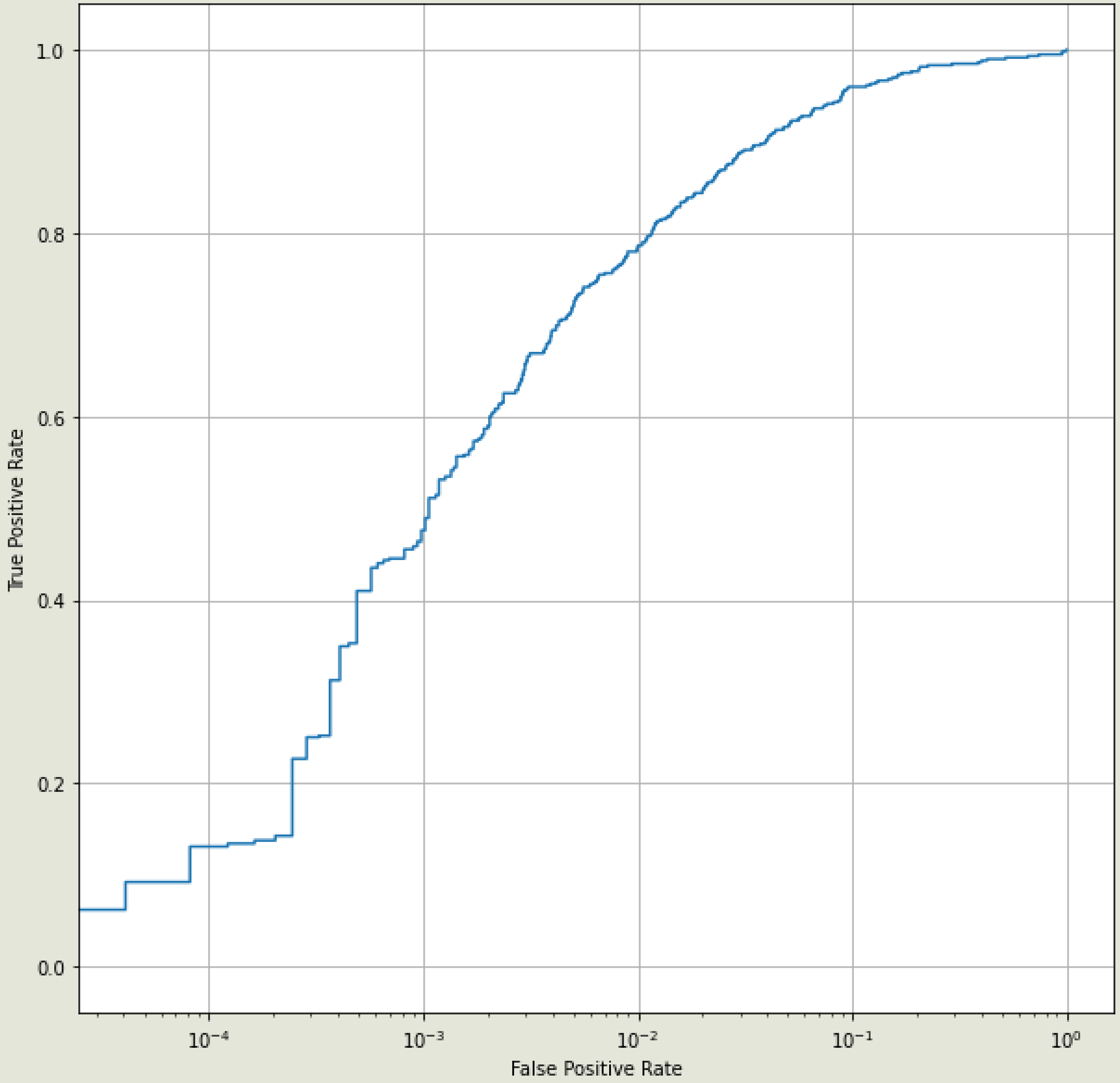


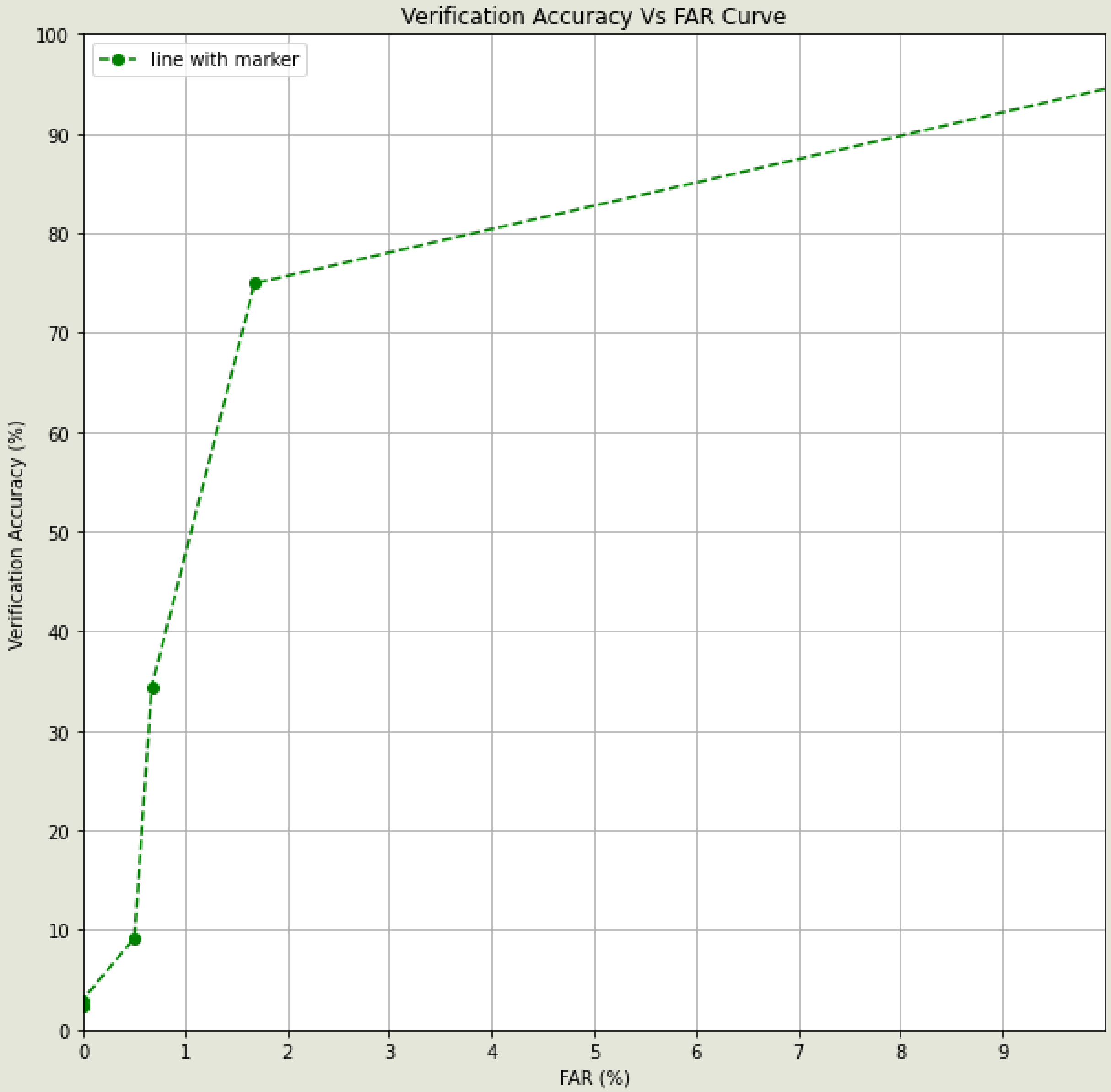
- **Cosine Augmented Center loss** (LAttCNN29 (Light Attention CNN 29))



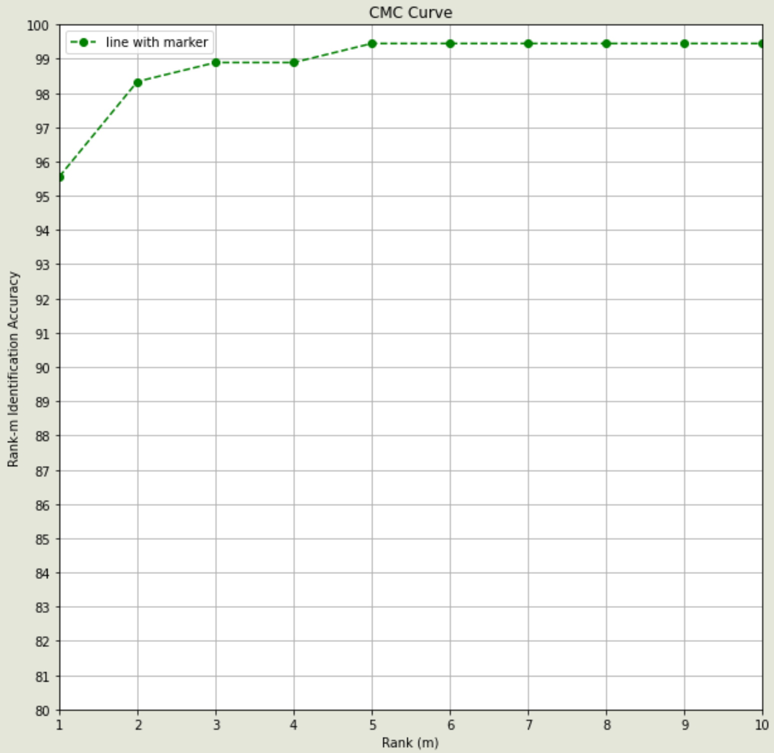


○ **ArcFace + Variance Loss** (LAttCNN29 (Light Attention CNN 29))





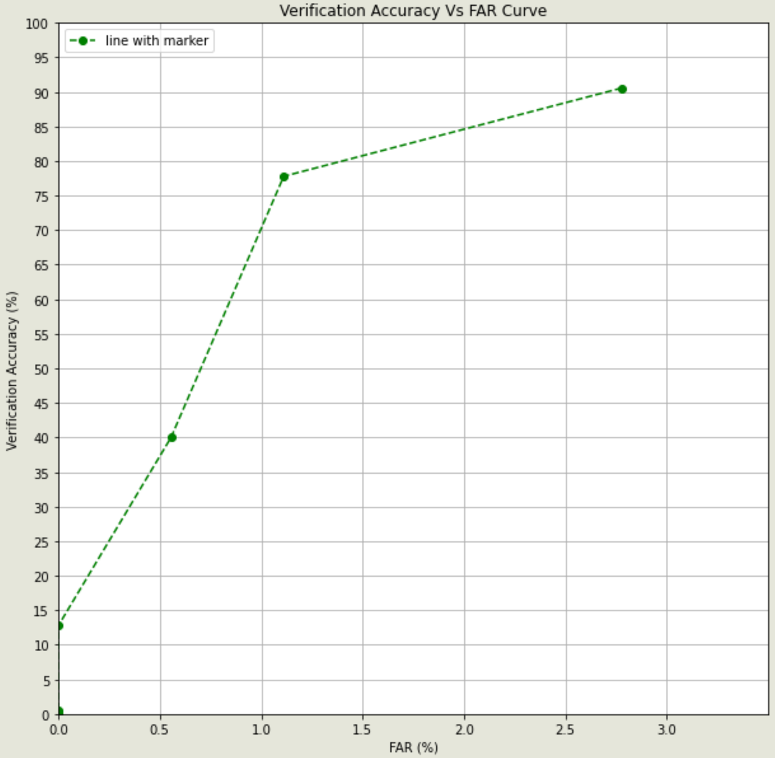
- **Using Plastic Surgery Face Database:** Fine-tuned models
    - **Contrastive** (LightCNN29)



Rank1 Identification accuracy for LightCNN29 using Pre-trained model after fine tuning with Contrastive Loss for Plastic Dataset(%) 95.55555555555556

Rank5 Identification accuracy for LightCNN29 using Pre-trained model after fine tuning with Contrastive Loss for Plastic Dataset(%) 99.44444444444444

Rank10 Identification accuracy for LightCNN29 using Pre-trained model after fine tuning with ContrastiveLoss for Plastic Dataset(%) 99.44444444444444



➢ Using LFW dataset:

LightCNN29 Pre-trained Init				
		Identification Accuracy		
Model	**Loss Function**	**Rank1**	**Rank5**	**Rank10**
LightCNN29	ArcFace	54.60%	68.30%	71.80%
	Contrastive + ArcFace	75.60%	89.80%	93.30%
	Contrastive + Variance	90.60%	93.80%	94.70%
	SemiHard Triplet	72.00%	84.00%	88.80%
	SupCon with SNR Dist	86.60%	91.30%	92.80%
Compact Convolutional Transformer(CCT)	Contrastive + Variance	100%	100%	100%

Fine-tuned				
		Identification Accuracy		
Model	**Loss Function**	**Rank1**	**Rank5**	**Rank10**
LightCNN29	Contrastive + Variance	96.40%	98.30%	98.50%
Compact Convolutional Transformer(CCT)	Contrastive + Variance	4%	13.40%	19.10%

➢ Using Plastic Surgery Face Database:

Fine-tuned					
		Identification Accuracy			Verification Accuracy
Model	**Loss Function**	**Rank1**	**Rank5**	**Rank10**	**1% FAR**
LightCNN29	Contrastive	95.55%	99.44%	99.44%	70%

➢ Using DFW dataset:

Fine-tuned		
		Verification Accuracy
Model	**Loss Function**	**1% FAR**
LightCNN29	Contrastive	49%
LAttCNN29 (Light Attention CNN 29)	Center + Angular	45%
LAttCNN29 (Light Attention CNN 29)	Cosine Augmented Center loss	23%
LAttCNN29 (Light Attention CNN 29)	ArcFace + Variance Loss	49%

Conclusion

- ➢ The datasets LFW, Plastic Surgery Faces and DFW (Protocol1) are small sample datasets.
- ➢ Training from scratch or using pre-trained initializations of LightCNN29 didn't give good results.
- ➢The loss function Cosine Augmented Center Loss, didn't give satisfactory results.
- ➢The best results were found by fine-tuning pre-trained LightCNN29 model updated with Attention Modulated MFM, using Contrastive Loss and variants.
- ➢Further exploring architectures such as DenseNet or ViT for fine-tuning along with loss function updates can give better results (rank1 accuracy & verification accuracy).

THANK YOU!