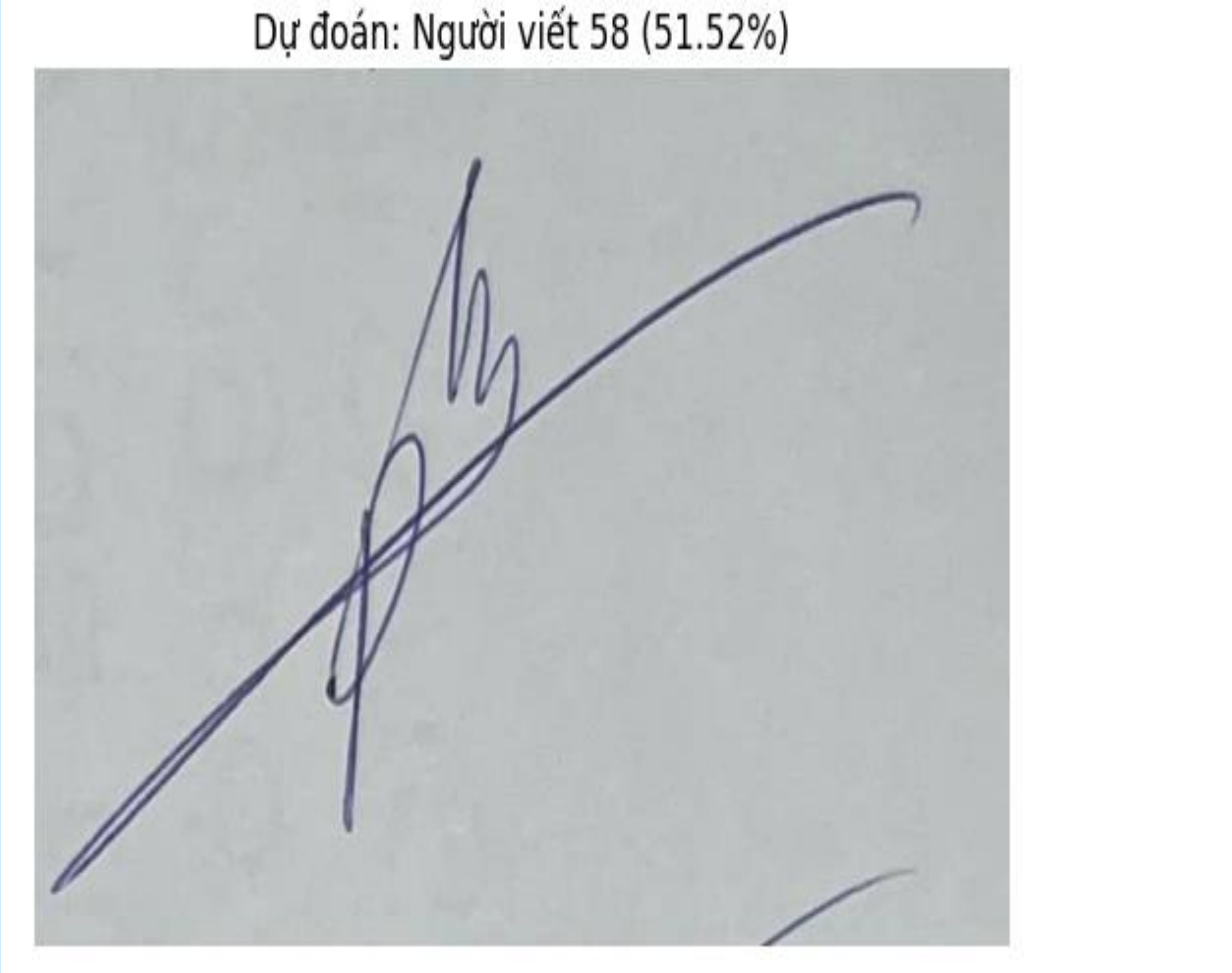
Solving Signature Security Challenges with CNNs

Writer Identification vs. Signature Verification

Presented by: Tran Huu Dat

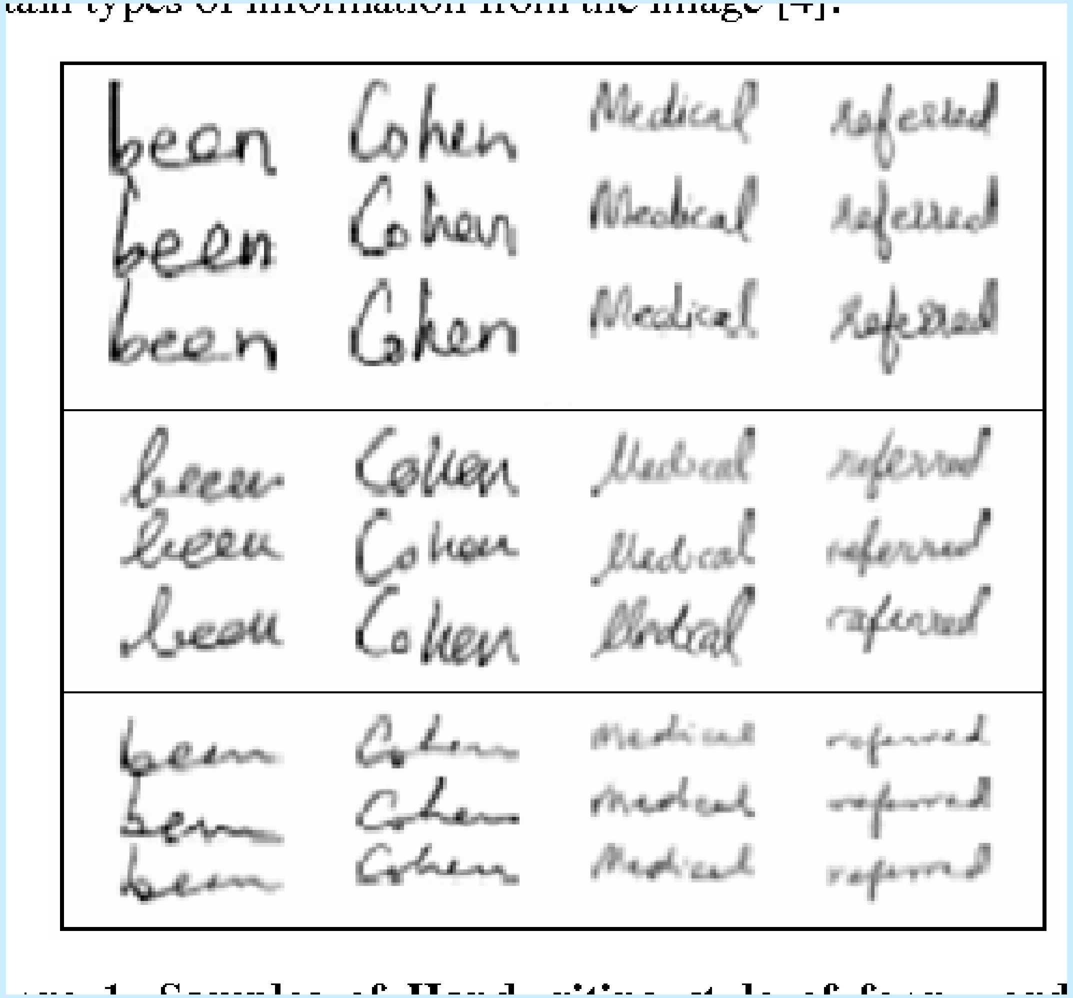
Two Problems, Two Questions

Identification (Classification)



Who is this person?

Verification (Metric Learning)



Are they the same person?

Approach 1: Writer Identification (Classification)

Architecture

Standard CNN (Conv -> Pool -> Dense)

Learning Mechanism

Sparse Categorical Cross Entropy

Pros

Straightforward, easy to implement.

Cons

Not scalable (requires retraining for new users), struggles with unseen users.

Architecture Explained

Standard CNN (Conv -> Pool -> Dense)

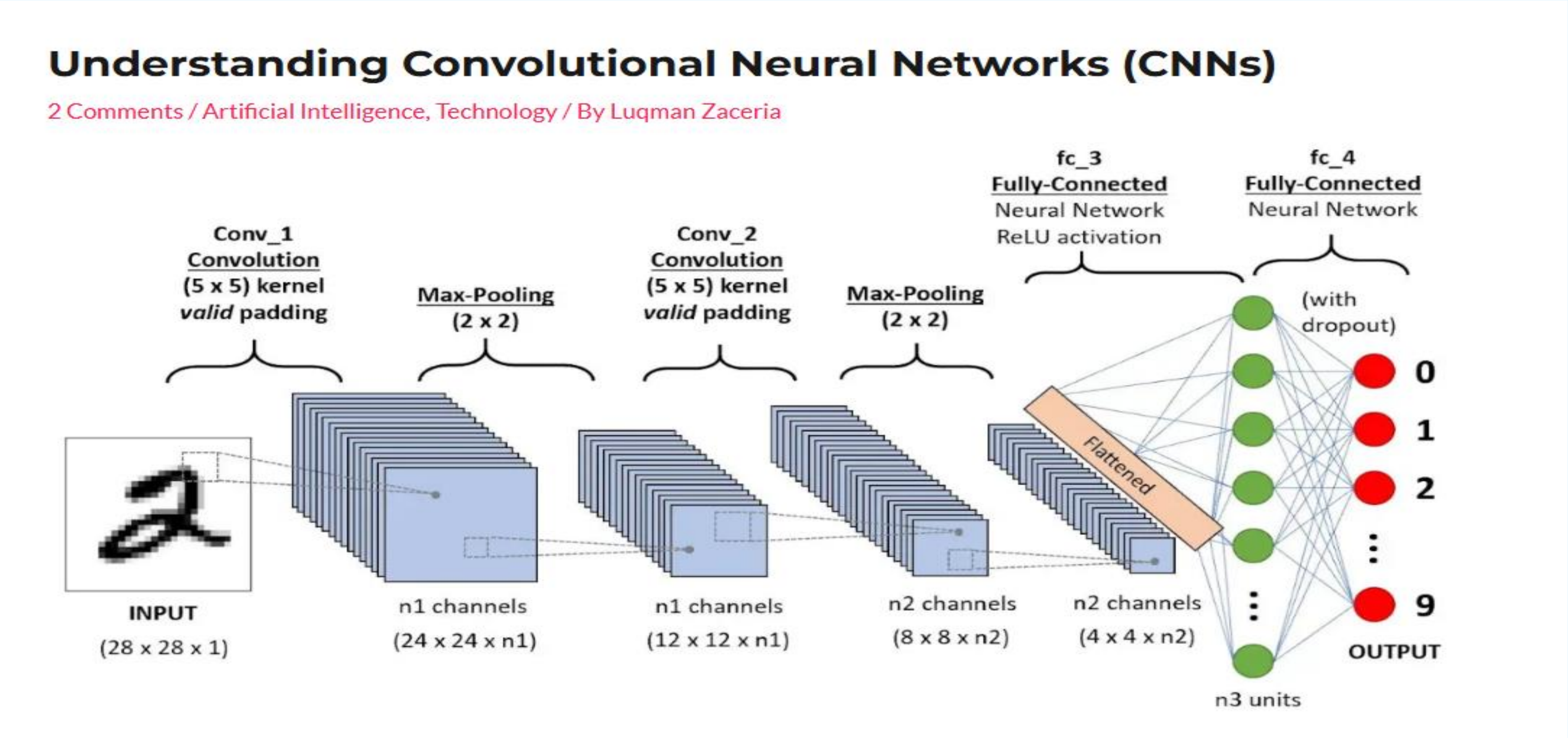
- Convolutional (Conv) layers act as feature detectors, automatically learning to identify critical patterns like lines, curves, and textures in the signature.
- Pooling (Pool) layers reduce the data's spatial dimensions, making the model more efficient and robust to small variations in the signature's position.
- The final Dense layers synthesize all the learned features to make the final classification decision: identifying the writer's ID.

Learning Mechanism Explained

Sparse Categorical Cross Entropy

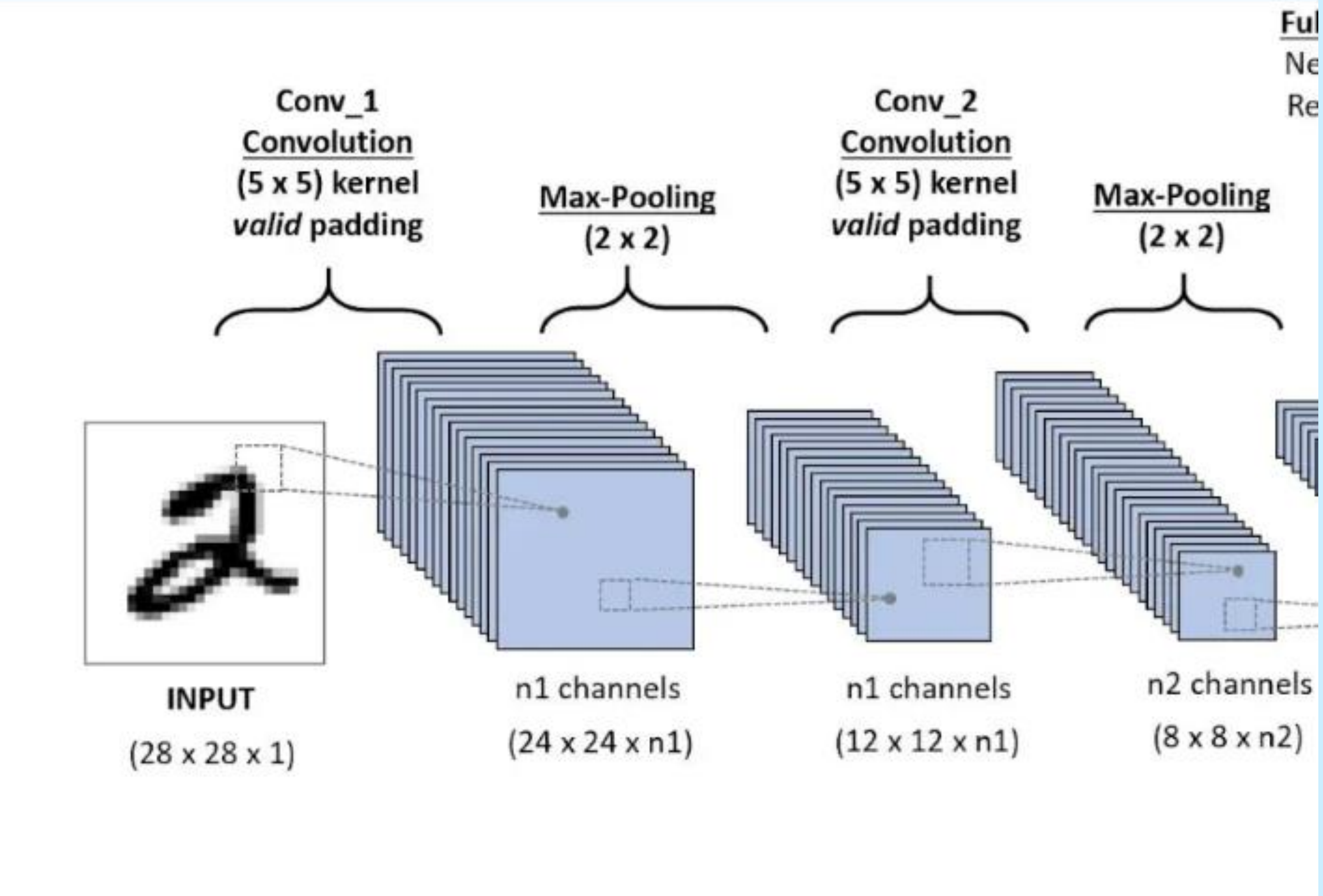
- This is a loss function specifically designed for multi-class classification problems where the labels are single integers.
- It measures the difference between the model's predicted probability distribution and the actual writer's ID. The training process aims to minimize this error.

Approach 1: Writer Identification (Classification)



Source: https://lumos.blog/understanding-convolutional-neural-networks-cnns/

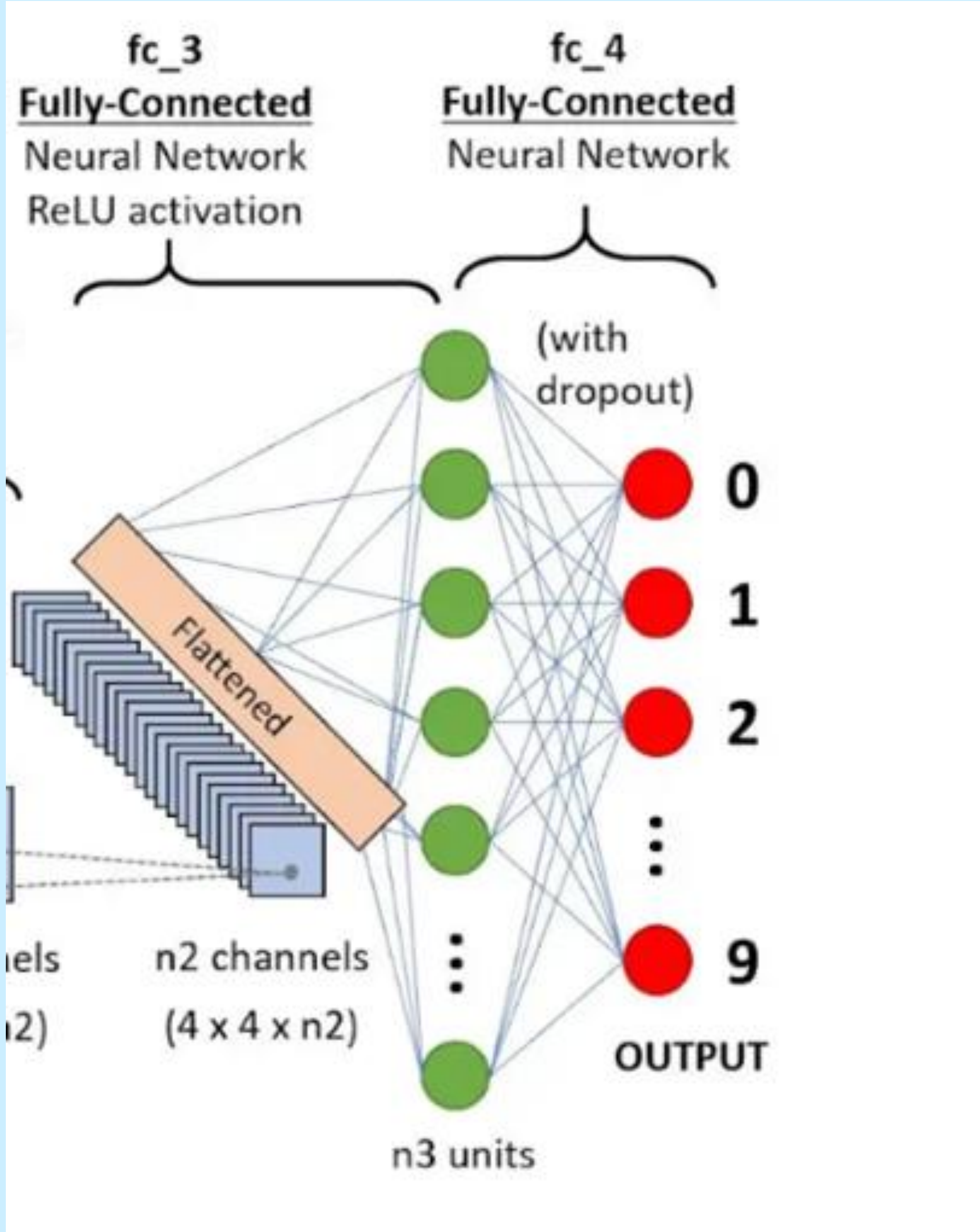
Step 1: Feature Extraction (Conv + Pool)



This initial stage is the core of the CNN. The network scans the image multiple times to build a rich map of increasingly complex features.

Source: https://lumos.blog/understanding-convolutional-neural-networks-cnns/

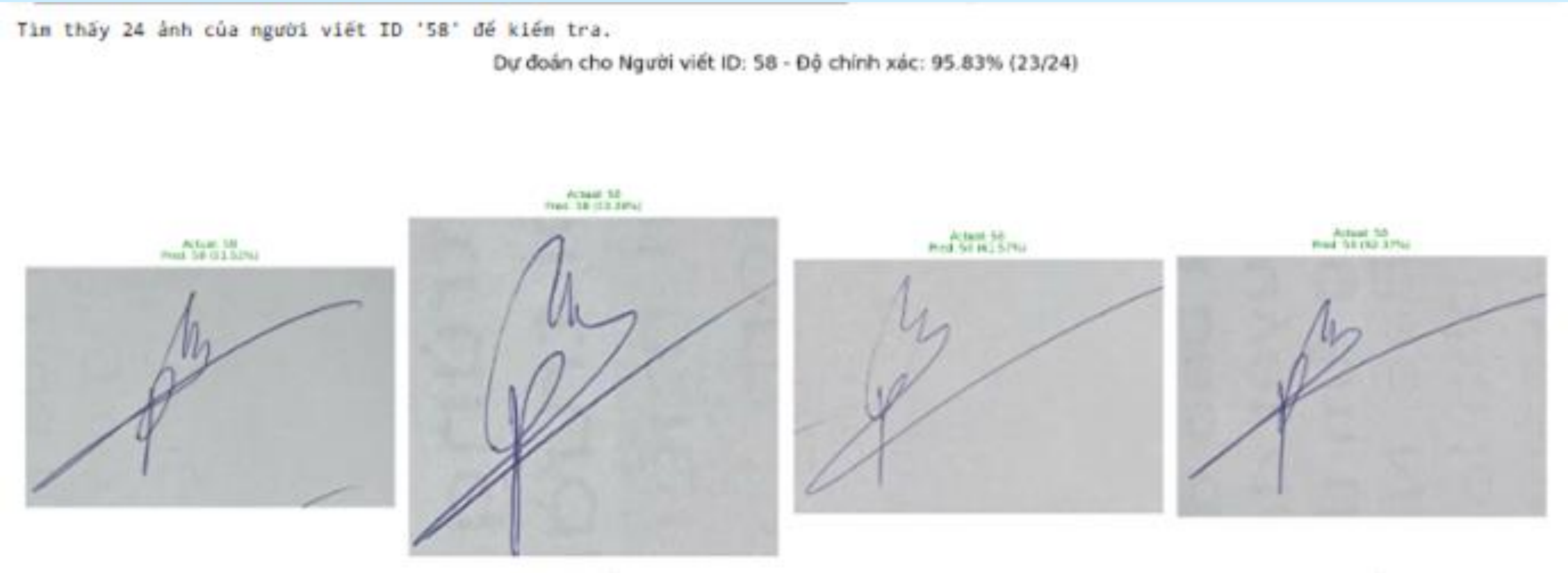
Step 2: Classification (Flatten + Dense)



After feature extraction, the 2D feature map is "flattened" into a 1D vector.
The Fully-Connected layers then analyze this vector to make the final prediction.

Source: https://lumos.blog/understanding-convolutional-neural-networks-cnns/

LIVE DEMO - WRITER IDENTIFICATION



Link demo: [(25-10-2025) SIGNATURES.ipynb]

Approach 2: Signature Verification (Metric Learning)

1. **Architecture**

   Siamese Network (Shared Weights)

2. **Learning Mechanism**

   Triplet Loss / Contrastive Loss

3. **Pros: Scalable**

   (no retraining for new users), more robust.

4. **Cons:**

   More complex to implement, training can be tricky.

Architecture Explained: Siamese Network

Two identical CNN "towers" with Shared Weights.

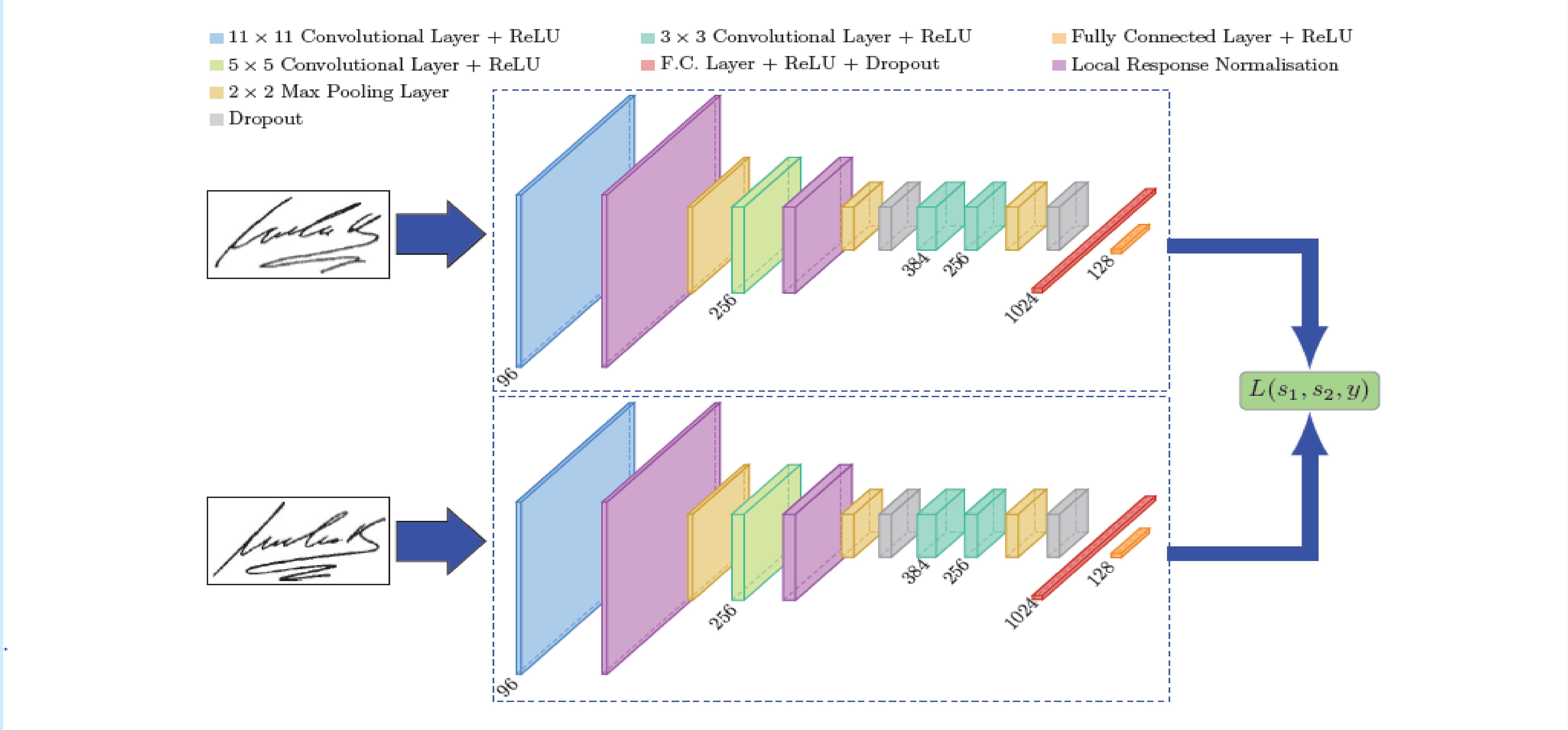
- Siamese" means twin. This architecture uses two identical CNN networks that, crucially, share the exact same weights.
- This ensures that both input images are processed and mapped into the feature space in the exact same way, creating a fair basis for comparison.

Learning Mechanism Explained: Triplet Loss

Anchor - Positive - Negative

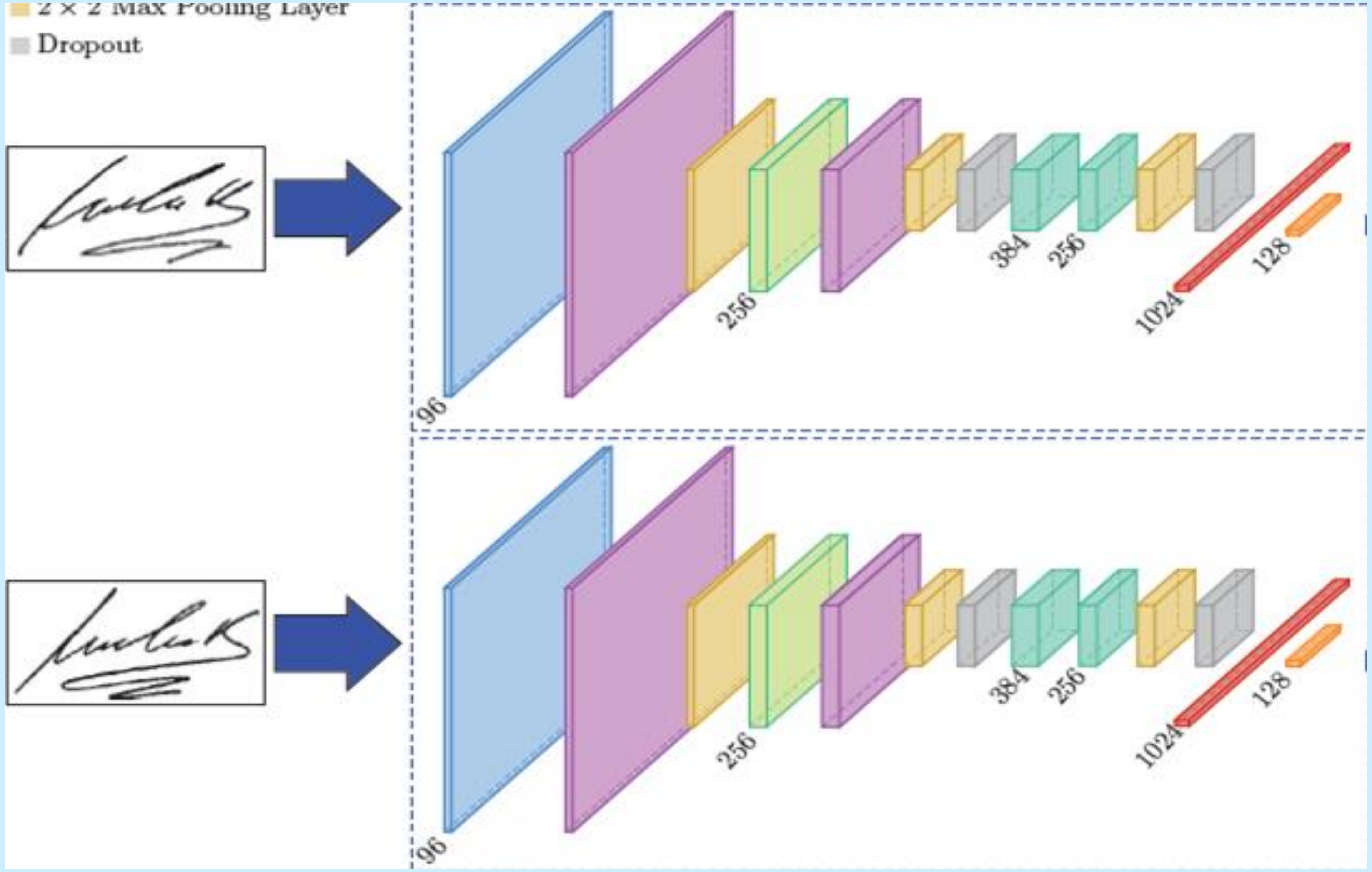
- The model learns by comparing three images simultaneously: an Anchor (the base signature), a Positive (another signature from the same person), and a Negative (a signature from a different person).
- The goal of the loss function is to "pull" the Anchor and Positive closer together while "pushing" the Anchor and Negative further apart in the feature space.

Approach 2: Signature Verification (Metric Learning)



Source: https://tiensu.github.io/blog/55_siamese_network/

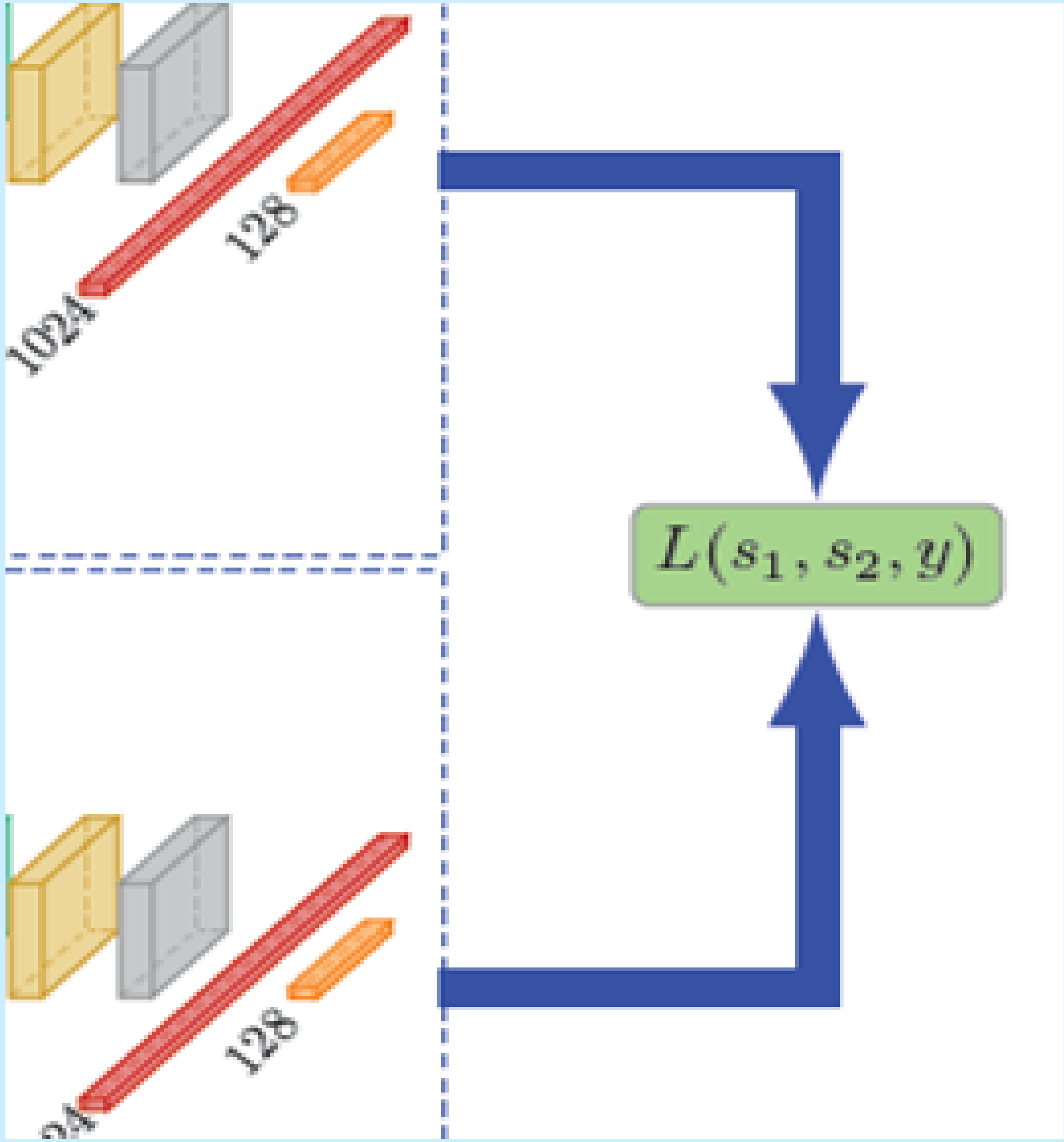
Approach 2: Signature Verification (Metric Learning)



The two signature images are passed through the parallel, identical CNN towers to generate two "feature vectors," which represent the essence of each signature.

Source: https://tiensu.github.io/blog/55_siamese_network/

Approach 2: Signature Verification (Metric Learning)



Instead of classifying, the model calculates the "distance" or similarity between the two feature vectors. The training process optimizes the network's weights so this distance accurately reflects the real-world relationship (same or different writer).

Source: https://tiensu.github.io/blog/55_siamese_network/

Comparing the Two Approaches

Feature	Identification (My Demo)	Verification (Siamese)
Goal	Classify among known users	Match a pair of signatures
Scalability	Poor (requires retraining)	Excellent
Use Case	Forensic analysis (closed set)	Banking, Access Control (open set)
Complexity	Simpler	More Complex

Conclusion

Main Takeaway

The choice of CNN architecture depends critically on the specific problem definition.

Summary

Classification (Identification) is powerful but limited, while Metric Learning (Verification) offers scalability and flexibility for real-world systems.