Exercise metric learning : find your doppleganger

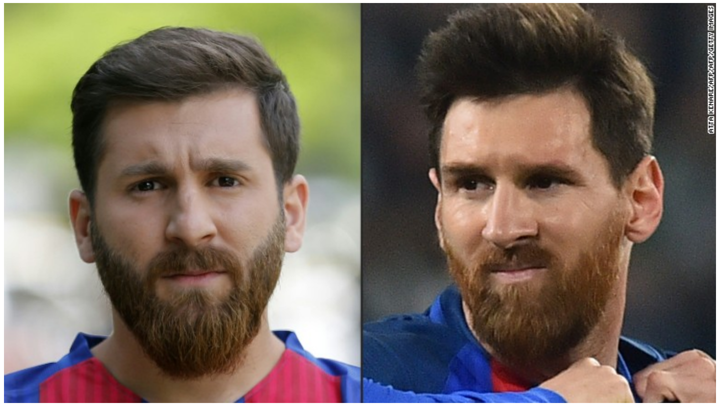
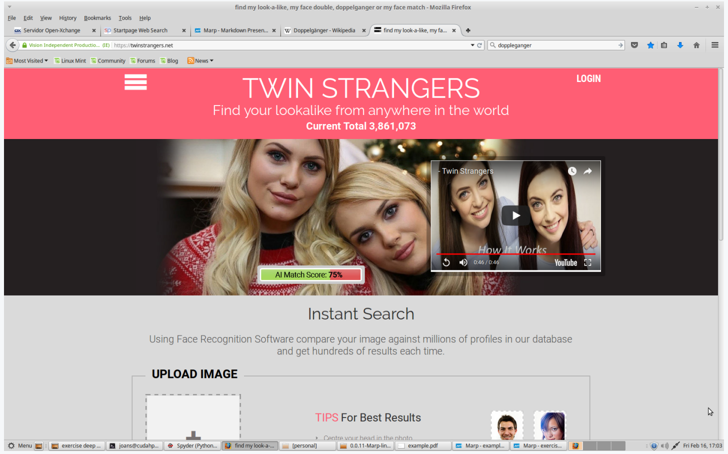
Module 5, Master Computer Vision, 2018-19

Joan Serrat

http://www.cvc.uab.es/people/joans/

What's a doppleganger ?

A person who closely reseembles another but is unrelated to him/her. Also, a double and a look-alike. Can be a living person, a portrait in a museum ...

	Iranian doppleganger of Leo Messi
 find my look-a-like, my face double, doppelganger or my face match - Mozilla Firefox TWIN STRANGERS Find your lookalike from anywhere in the world Current Total 3,861,073 AI Match Score: 75% Instant Search Using Face Recognition Software compare your image against millions of profiles in our database and get hundreds of results each time. UPLOAD IMAGE TIPS For Best Results Centre your head in the photo	https://twinstrangers.net/

How to find him/her ?

- define a **metric**, ie. a measure of similarity
- or better, **learn it** from samples
- samples = **pairs of images** from one the same person
- dataset with **many persons, many images** per identity
- once learned, provide an image and find the most similar instance in the dataset

This is impossible with the available data and time. So we'll just do our best on a small dataset

It won't be a real look-alike, but hopefully will somehow ressemble you!

The *real* objectives of the exercise

- Learn **by example** how to :
  - build a siamese/triplet network
  - draw pairs of samples to train it
  - make a simple distance-based classifier

  in **TensorFlow**, **Keras** or **PyTorch**, as you prefer
- Implement a **loss** function
- Evaluate the effect of some **mining** technique on the training and accuracy
- **Visualize** the embedding with t-SNE
- Perform and assess **image retrieval**

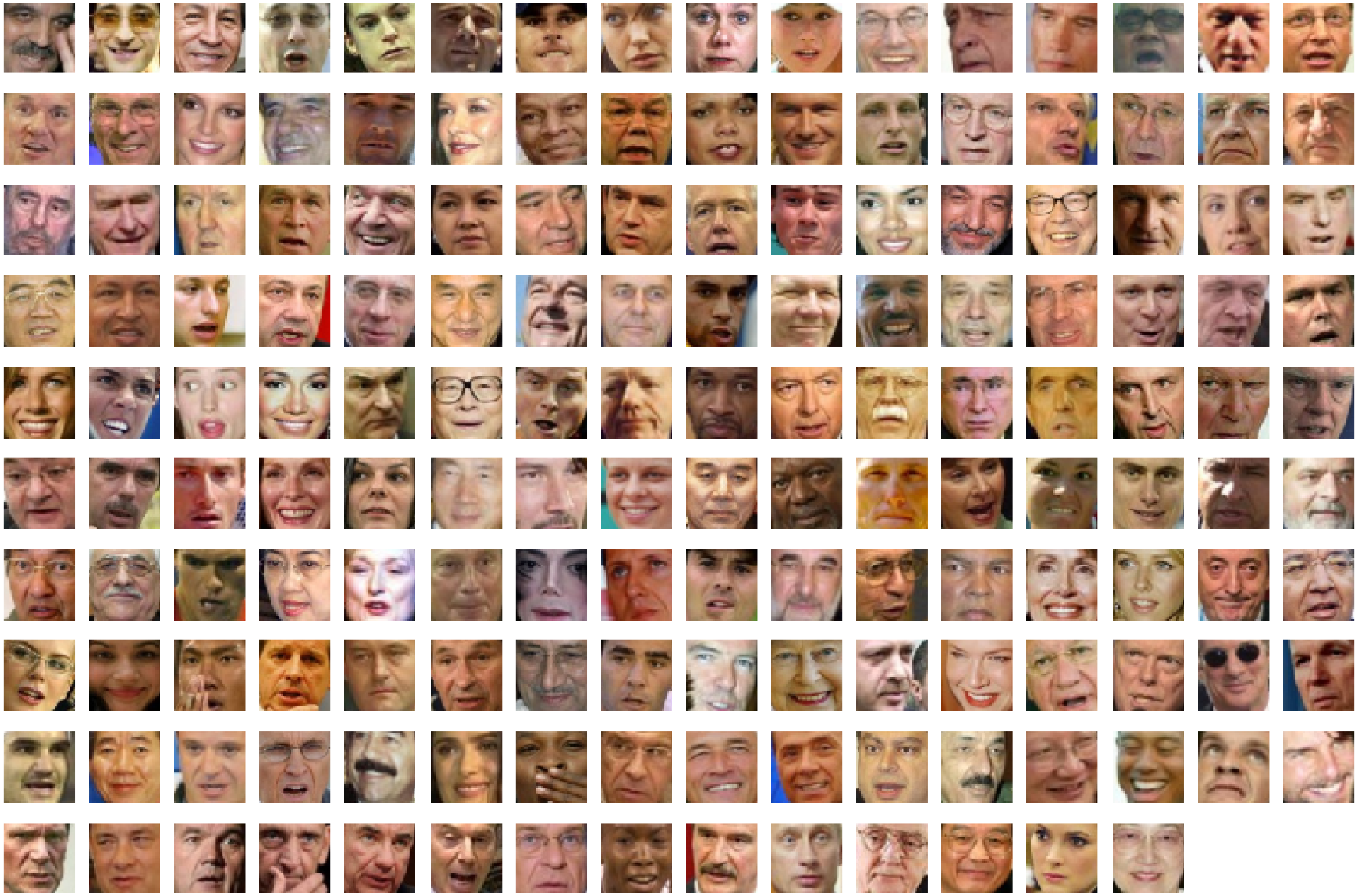
Datasets : Labeled Faces in the Wild

- +5K identities of celebrities, sportsmen, politicians ... (back in 2009)
- higly variable number of images per identity
- only 62 identities with > 20 images and 168 with > 10 images
- **croped face region** to avoid influence of background
- color, **resized** 64 x 64

Datasets : Labeled Faces in the Wild

62 identities

Datasets : Labeled Faces in the Wild

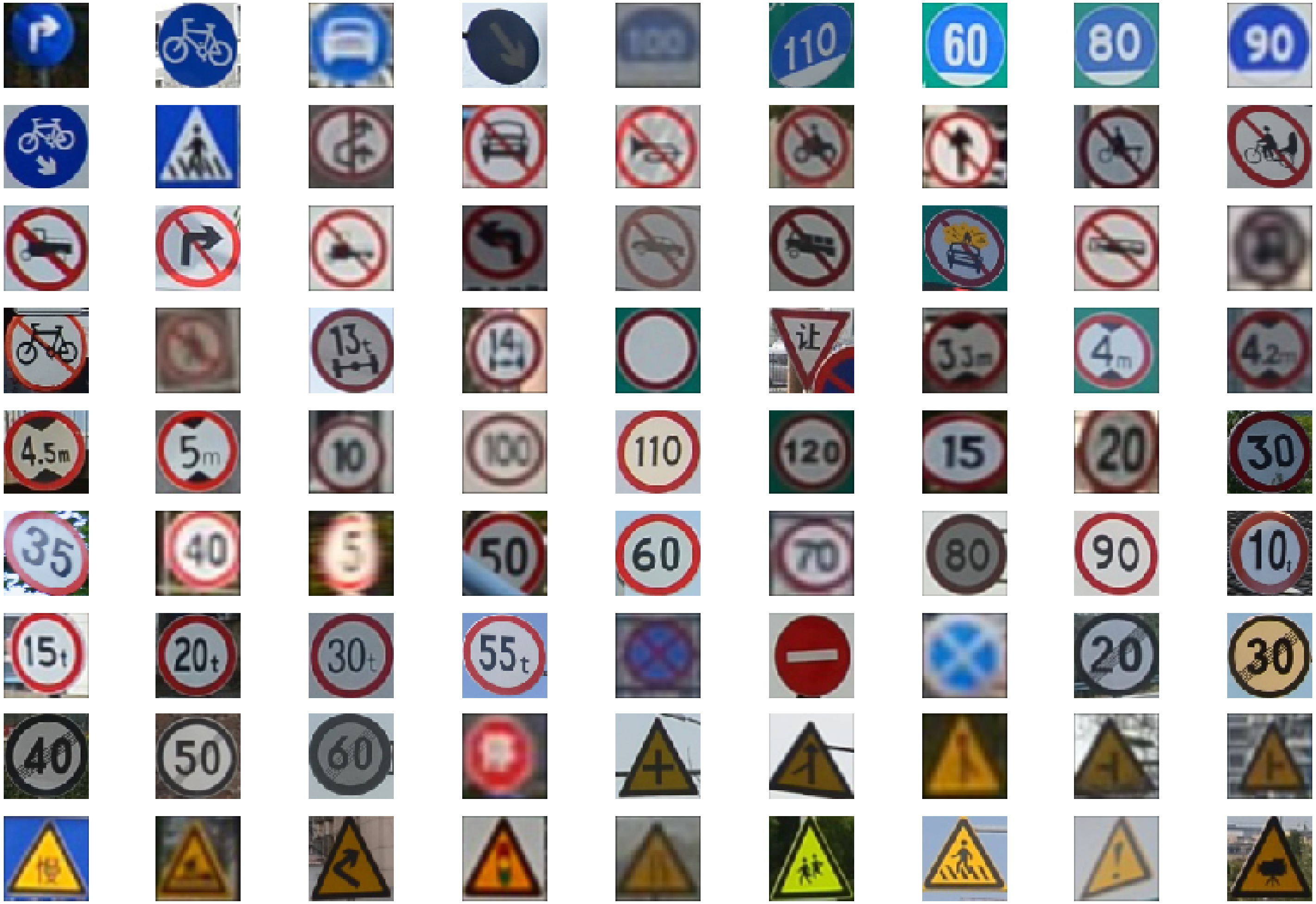


168 identities

Datasets : Tsinghua Traffic Signs

- +170 traffic signs from China
- diverse camera orientation, changing lighting, occlusions
- croped from the original frames and resized 64 x 64
- only 81 traffic signs with > 20 images

Datasets : Tsinghua Traffic Signs



81 traffic signs

What to do

1. Learn a **baseline** metric for Tsinghua and LFW for a dimension of 2 (just for visualization) and then 16, 32 or 64
2. Compute classification **accuracy** in Tsinghua with some distance-based classifier : nearest class mean, k-nearest neighbors (implemented in scikit-learn.org)
3. **Visualize** the training and testing set samples on the 2d and nd embedding spaces in Tsinghua and comment the results. Use PCA or t-SNE in TensorBoard or scikit-learn.org
4. **Do retrieval** : find your k doppleganger candidates in LFW as k-NN ($k = 1$ is ok)
5. Change the contrastive loss to **triplet loss**
6. Do *some kind* of **mining** and compare with baseline in terms of accuracy and evolutions of the loss. Tensorflow has some already implemented.

What to do

This is not a challenge to achive the best accuracy.

I've got 95% in Tsinghua (81 classes) and just 83% (62 classes) in LFW

It's more about how to

- learn an embedding and visualize it
- do mining
- use it for classification and explain the results

Code

We provide a lot of code ready to use :

- siamese based on VGG-16
- generation of batches of pairs for both Tsinghua and LFW
- split datasets into train / validation / test
- accuracy and nearest neighbor computation
- visualization in 2d and nd with scikit-learn t-SNE (but TensorBoard implementation is much cooler)
- one tower + siamese loss, hint on triplet loss

Code + datasets here

Grading

B = 0.25 exam points if you do the basics : 1 to 4

A = 0.5 points if B + 5 or B + 6

Deliverables

Source code and at most 10 slides in PDF packed in a zip file to be sent to joans@cvc.uab.es by April 2nd, 23:59.

This is team work. Tell me who did what in the last slide.