Face Recognition

Shusen Wang

Training

Challenge of Face Recognition

Question: Why not using Softmax classifier (as in image recognition)?

- Softmax classifier is a dense layer with Softmax activation.
- #class is large (can be millions or billions).
- #parameters in the output layer is huge!
  - Suppose the input shape of Softmax classifier is $1K$.
  - Suppose #class is $10M$.
  - Then #parameters $= 1K \times 10M = 10G$.

Randomly Sample a Triplet

positive $\mathbf{x}^+$

anchor $\mathbf{x}^a$

negative $\mathbf{x}^-$

Feature Extraction using CNN

positive $\mathbf{x}^+$  $\mathbf{f}$ $\mathbf{f}(\mathbf{x}^+)$

anchor $\mathbf{x}^a$  $\mathbf{f}$ $\mathbf{f}(\mathbf{x}^a)$

negative $\mathbf{x}^-$  $\mathbf{f}$ $\mathbf{f}(\mathbf{x}^-)$

Loss Functions

positive $\mathbf{x}^+$

$\mathbf{f}$

$\mathbf{f}(\mathbf{x}^+)$

anchor $\mathbf{x}^a$

$\mathbf{f}$

$\mathbf{f}(\mathbf{x}^a)$

negative $\mathbf{x}^-$

$\mathbf{f}$

$\mathbf{f}(\mathbf{x}^-)$

$$\left\|\mathbf{f}(\mathbf{x}^+) - \mathbf{f}(\mathbf{x}^a)\right\|_2^2$$

$$\left\|\mathbf{f}(\mathbf{x}^a) - \mathbf{f}(\mathbf{x}^-)\right\|_2^2$$

Loss Functions

- N triplets: $(\mathbf{x}_1^{\mathrm{a}}, \mathbf{x}_1^{+}, \mathbf{x}_1^{-}), (\mathbf{x}_2^{\mathrm{a}}, \mathbf{x}_2^{+}, \mathbf{x}_2^{-}), \cdots, (\mathbf{x}_N^{\mathrm{a}}, \mathbf{x}_N^{+}, \mathbf{x}_N^{-})$.
  - N can be far larger than n (#samples).
- Loss function (option 1):

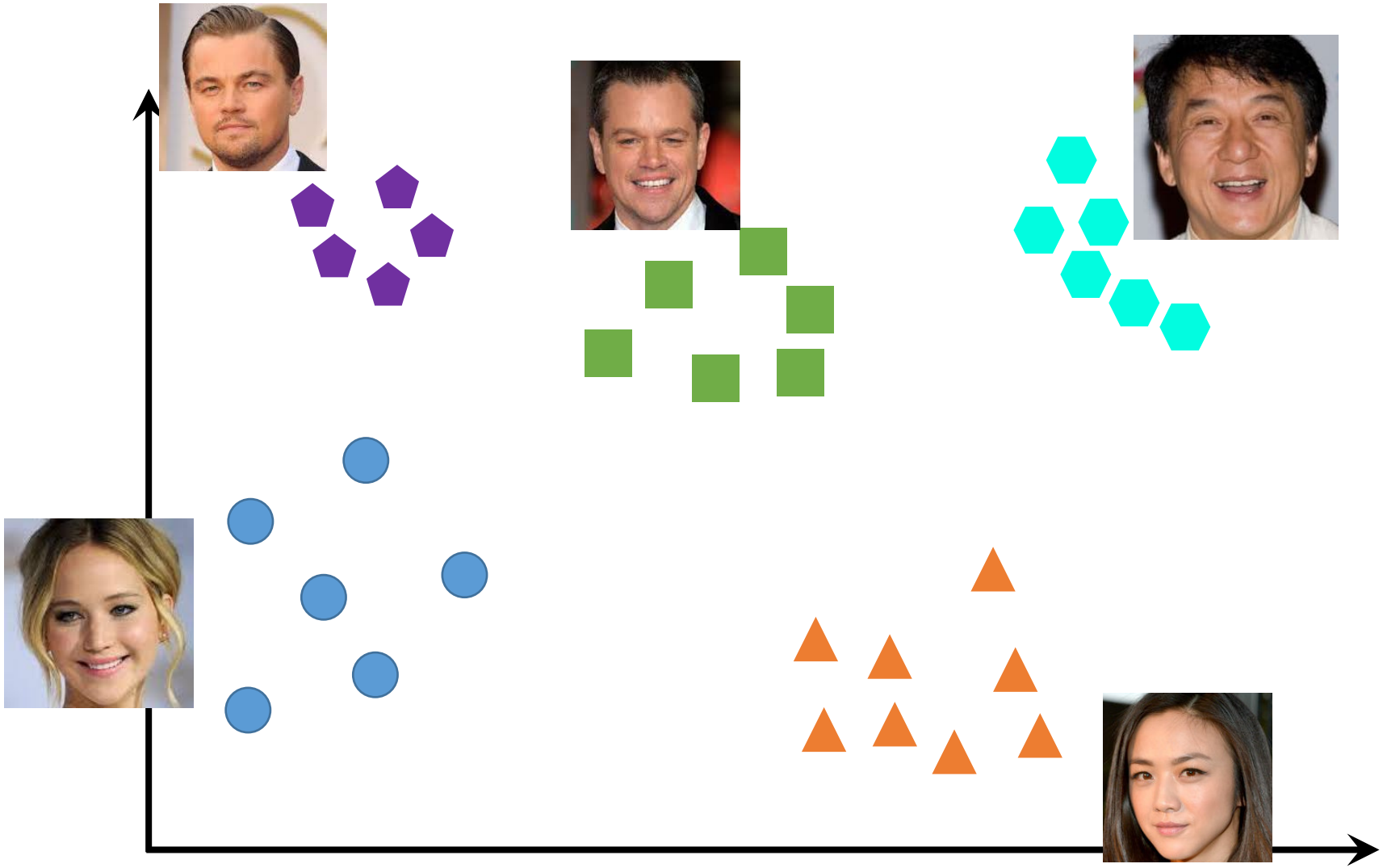
$$\frac{1}{N}\sum_{i=1}^{N}\left[\left|\left|\mathbf{f}(\mathbf{x}_i^{+})-\mathbf{f}(\mathbf{x}_i^{\mathrm{a}})\right|\right|_2^2-\left|\left|\mathbf{f}(\mathbf{x}_i^{\mathrm{a}})-\mathbf{f}(\mathbf{x}_i^{-})\right|\right|_2^2\right].$$

- Loss function (option 2):

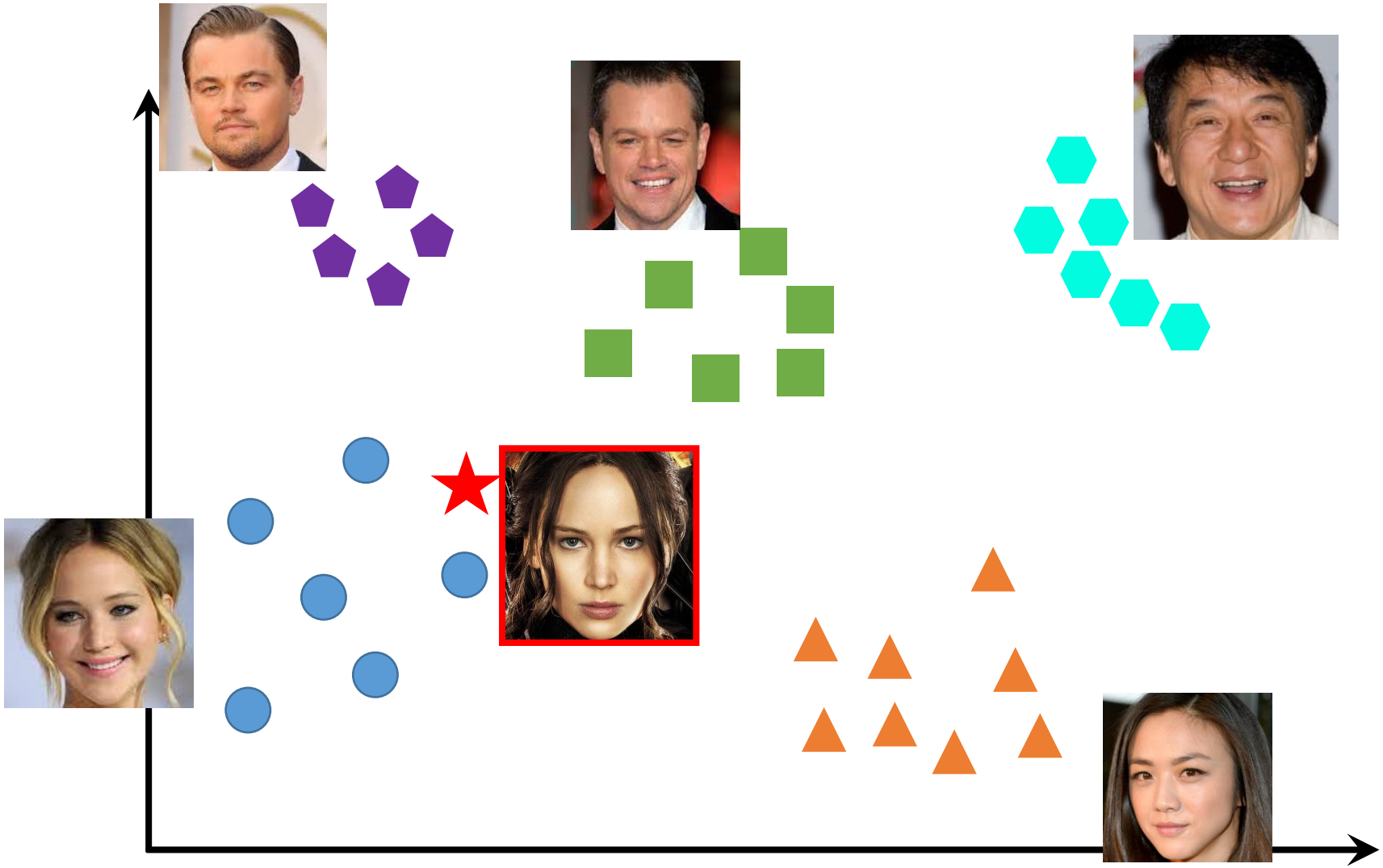
$$\frac{1}{N}\sum_{i=1}^{N}\left[\left|\left|\mathbf{f}(\mathbf{x}_i^{+})-\mathbf{f}(\mathbf{x}_i^{\mathrm{a}})\right|\right|_2^2-\left|\left|\mathbf{f}(\mathbf{x}_i^{\mathrm{a}})-\mathbf{f}(\mathbf{x}_i^{-})\right|\right|_2^2+\xi\right]_+.$$

Prediction

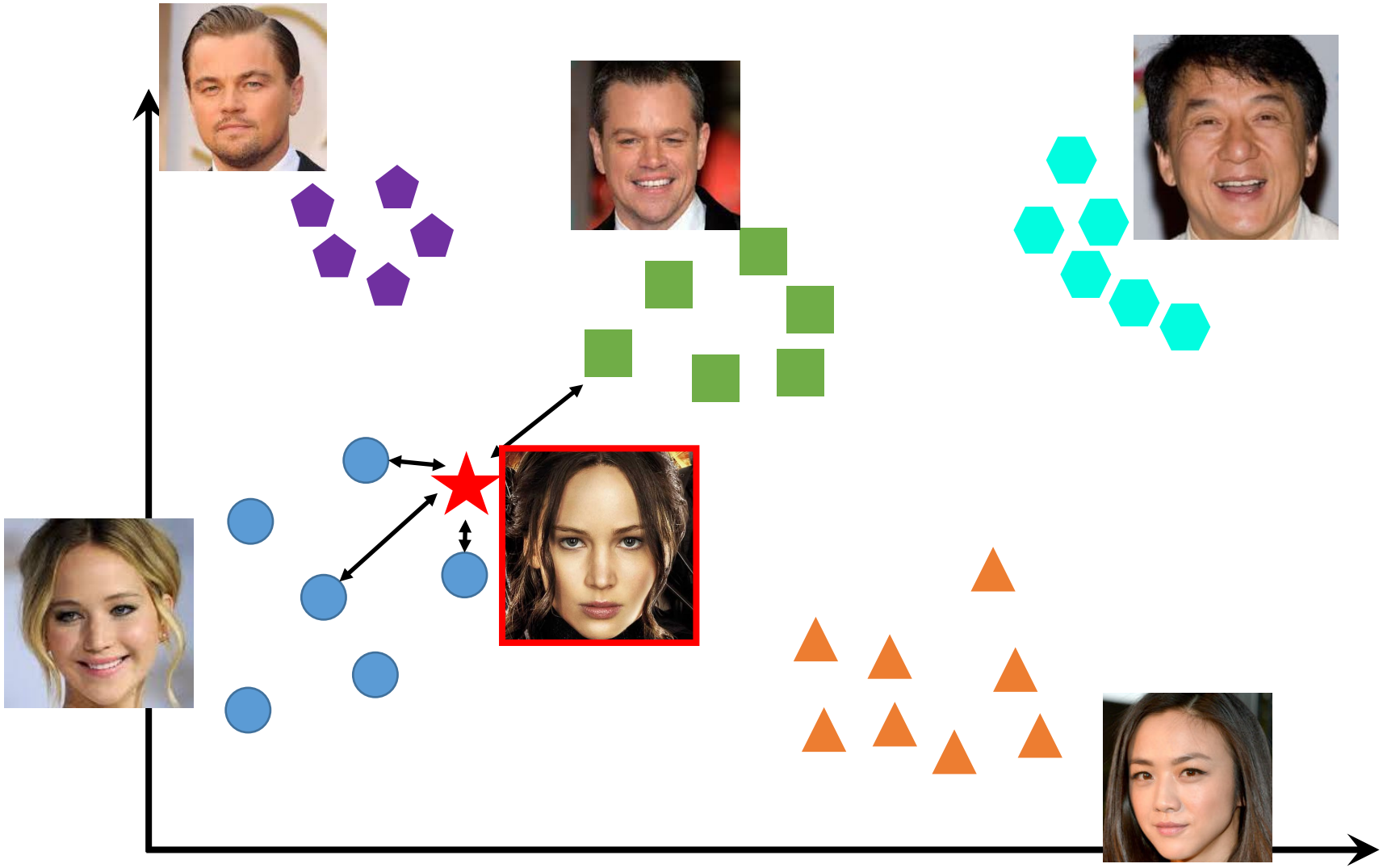
1. Feature Extraction for All the Training Photos



2. Feature Extraction for Test Sample



3. KNN Classifier



Keras Implementation

- http://krasserm.github.io/2018/02/07/deep-face-recognition/