WEEK 4

(c4)

• FACE RECOGNITION

+ Liveness detection

* Face verification v/s face recognition

Face verification	Face recognition
Input: image, id	Database of K persons
output: input image is of claimed person?	Input: image
	output: ID (person belonging to K)

* One shot learning

(Just one training example)

→ SIMILARITY FUNCTION

$d(img1, img2)$ = degree of difference b/w image

If $d(img1, img2) \leq \tau$ .... same person

$> \tau$ ... different people

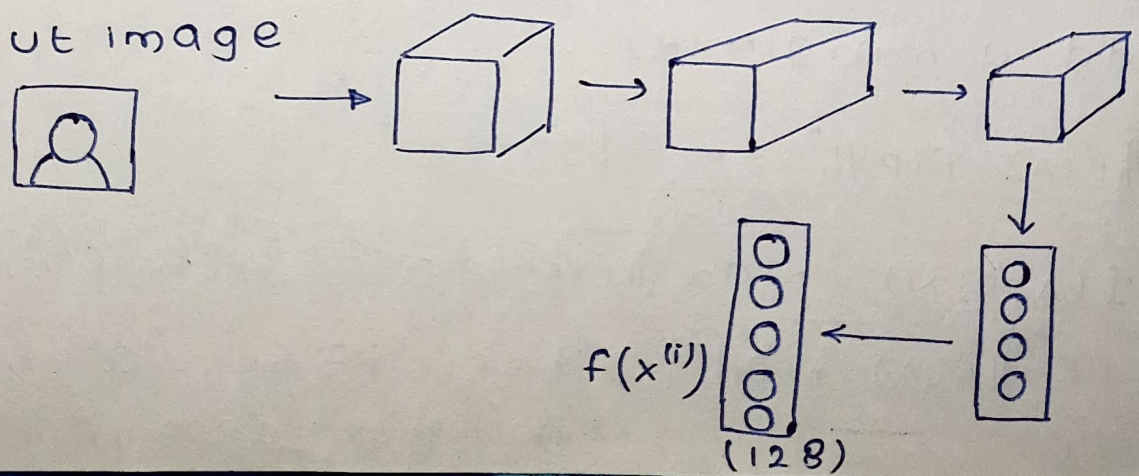
[τ = small threshold]

* Siamese network

$$d(x^{(1)}, x^{(2)}) = \| f(x^{(1)}) - f(x^{(2)}) \|^2$$

Run each input image on an independent neural network

eg: Input image



⇒ Parameters of ~~encoding~~ NN define an encoding $f(x^{(i)})$

Learn parameters so that:

- If $x^{(i)}, x^{(j)}$ are same person, $\|f(x^{(i)}) - f(x^{(j)})\|^2$ is small
- If $(x^{(i)}), (x^{(j)})$ are different people then $\|f(x^{(i)}) - f(x^{(j)})\|^2$ is large

* Triplet Loss

α = margin (to push Anchor-positive & Anchor-negative pair further away from each other)

Anchor ⇒ Reference image (person 1)

Positive ⇒ Another image of person 1, closer to Anchor

Negative ⇒ Image of person 2, different from anchor

3 images: A, P, N

$$\underbrace{\|f(A) - f(P)\|^2}_{d(A,P)} \leq \underbrace{\|f(A) - f(N)\|^2}_{d(A,N)}$$

$$\|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 \leq 0 \quad \ldots f(img) = \vec{0}$$

Add a margin (α)

$$\therefore \boxed{\|f(A) - f(P)\|^2 + \alpha \leq \|f(A) - f(N)\|^2}$$

$$\therefore \boxed{L(A, P, N) = \max\left(\|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 + \alpha, 0\right)}$$

⇒ If $\|f(A) - f(P)\|^2 - \|f(A) - f(N)\|^2 + \alpha < 0$ then loss = 0

If —11— > 0 then loss = $\|f(A) - f(P)\|^2$ ... ✎

cost function

$$J = \sum_{i=1}^{m} \mathcal{L}(A^{(i)}, P^{(i)}, N^{(i)})$$

* training set must have multiple pictures of same person.

* choosing A, P, N

1) A, P, N if chosen randomly, $d(A,P) + \alpha \leq d(A,N)$ is easily satisfied.

2) choose triplets that are harder to train on. $d(A,P) \approx d(A,N) \quad \alpha <<<<$

Goal → use gradient descent to minimize cost function J

* Binary classification

$$\hat{y} = \sigma\left(\sum_{k=1}^{128} w_k \left| f(x^{(i)})_k - f(x^{(j)})_k \right| + b\right)$$

for χ^2 formula:

$$\frac{\left(f(x^{(i)})_k - f(x^{(j)})_k\right)^2}{f(x^{(i)})_k + f(x^{(j)})_k}$$

* use pairs of images:

Target label = 1 → same

= 0 → different

• NEURAL STYLE TRANSFER

content (C) + style (S) } Generated image (G)

→ Initial layers (NEURONS DETECT ↘ Lines, colours, simple shapes)
→ Deeper layers (NEURONS DETECT ↘ complex pattern, texture)

* cost function

$$J(G) = \alpha J_{content}(C,G) + \beta J_{style}(S,G)$$

DIFFERENT "HYPERPARAMETERS" TO DENOTE WEIGHTS

* Find generated image 'G'

(1) Initialize G randomly ⇒ G : 100 × 100 × 3

(2) use gradient descent to minimize J(G)

$$G := G - \frac{\partial}{\partial G} J(G)$$

* content cost function

(1) choose hidden layer l (middle layer) to compute cost

(2) use pre-trained convNet (VGG-19)

(3) $a^{[l](C)}$ & $a^{[l](G)}$ be activation of layer l
If $a^{[l](C)}$ & $a^{[l](G)}$ are similar, both images have similar content

$$J_{content}(C,G) = \frac{1}{4 \times n_H \times n_W \times n_C} \sum_{\text{all entries}} (a^{(C)} - a^{(G)})^2$$

• style cost function

→ using layer 'l' activation to measure style

→ define style as correlation between activations across channels

→ **style / Gram matrix**

$a^{[l]}_{i,j,k}$ = activation at (i,j,k) (H, W, C)

$G^{[l]} = n_c^{[l]} \times n_c^{[l]}$ dimensional

- $G^{[l](S)}_{kk'} = \sum_{i=1}^{n_H^{[l]}} \sum_{j=1}^{n_W^{[l]}} a^{[l](S)}_{ijk} \, a^{[l](S)}_{ijk'}$, $k, k' = 1, \ldots, n_c^{[l]}$

- $G^{[l](G)}_{kk'} = \sum_{i=1}^{n_H^{[l]}} \sum_{j=1}^{n_W^{[l]}} a^{[l](G)}_{ijk} \, a^{[l](G)}_{ijk'}$

$$J_{style}^{[l]}(S,G) = \left\| G^{[l](S)} - G^{[l](G)} \right\|^2$$

$$= \sum_k \sum_{k'} \left(G^{[l](S)}_{kk'} - G^{[l](G)}_{kk'} \right)^2 \times \frac{1}{\left(2 \times n_H^{[l]} \times n_W^{[l]} \times n_c^{[l]} \right)^2}$$

$$J_{style}(S,G) = \sum_l \lambda^{[l]} J_{style}^{[l]}(S,G)$$

$\lambda^{[l]}$ Hyperparameter