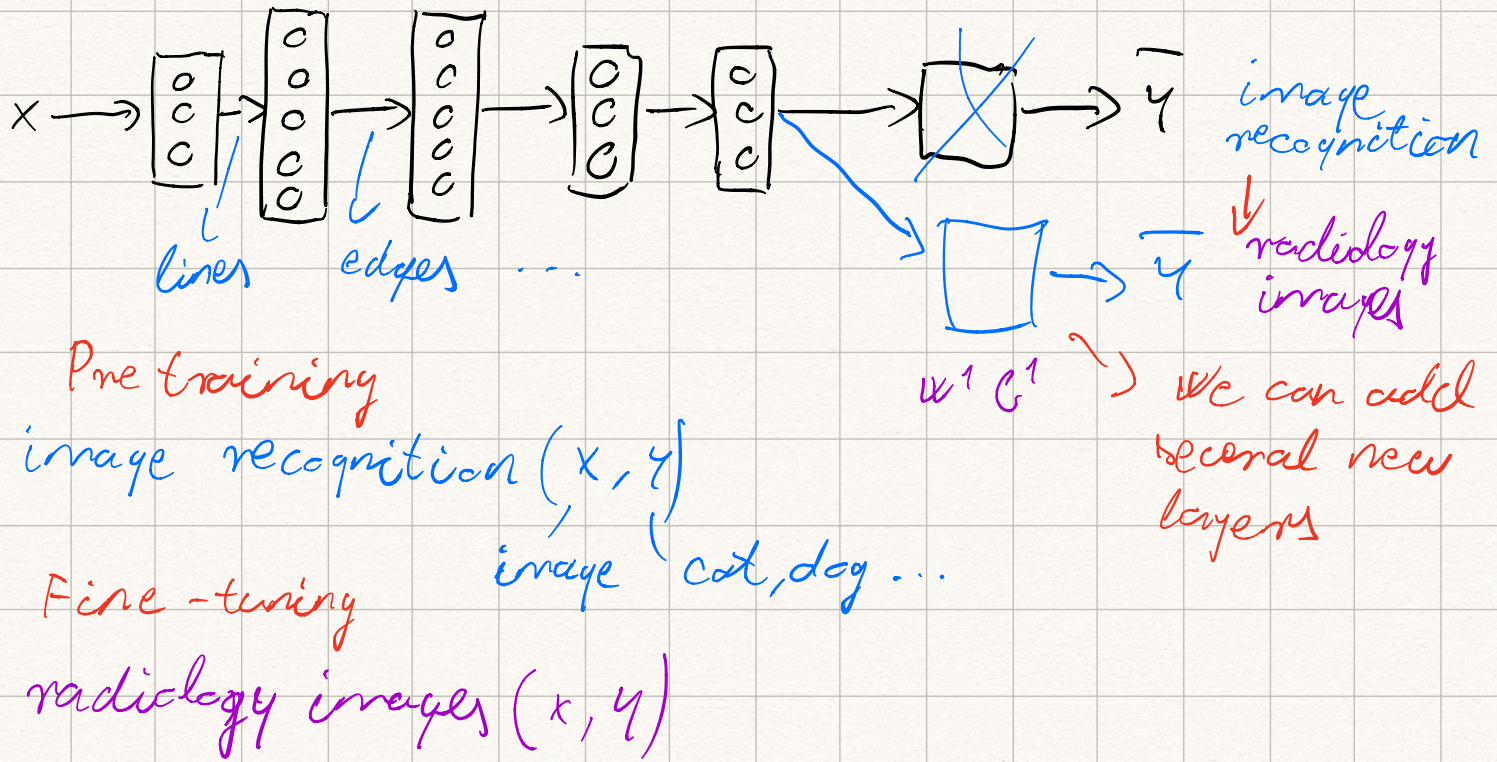


Transfer learning



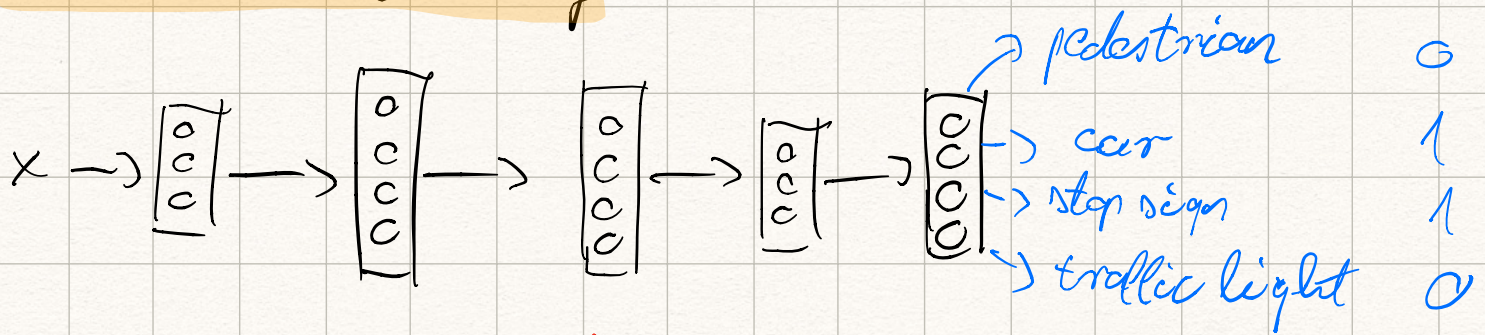
Transfer learning:

Learning lower features on large dataset and fine-tune the whole or only the last layers with specific dataset

Useful when:

- Task A and B have the same input. (images)
- You have lot more data for task A than task B
- low level features from A could be helpful for learning B

Multi-task learning



loss: $\frac{1}{m} \sum_{i=1}^m \left[\sum_{j=1}^4 y_j^i \right] \sigma \left(\frac{y_j^i}{y_j^i}, y_j^i \right)$

→ usual logistic reg.

sum only over
value of j with 0/1 label

unlike softmax regression

one image can have multiple labels

$$Y = \begin{bmatrix} 1 & 1 & 0 & \dots \\ c & c & 1 & \dots \\ c & c & ? & \dots \\ 1 & 0 & ? & \dots \end{bmatrix}$$

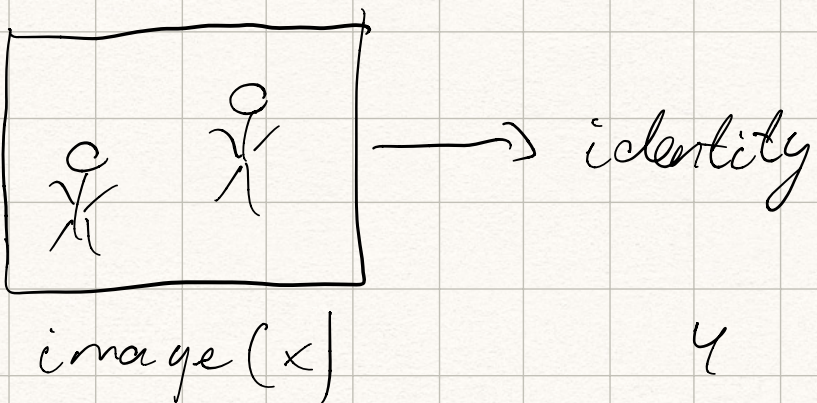
can be unlabeled data

multi-task learning makes sense:

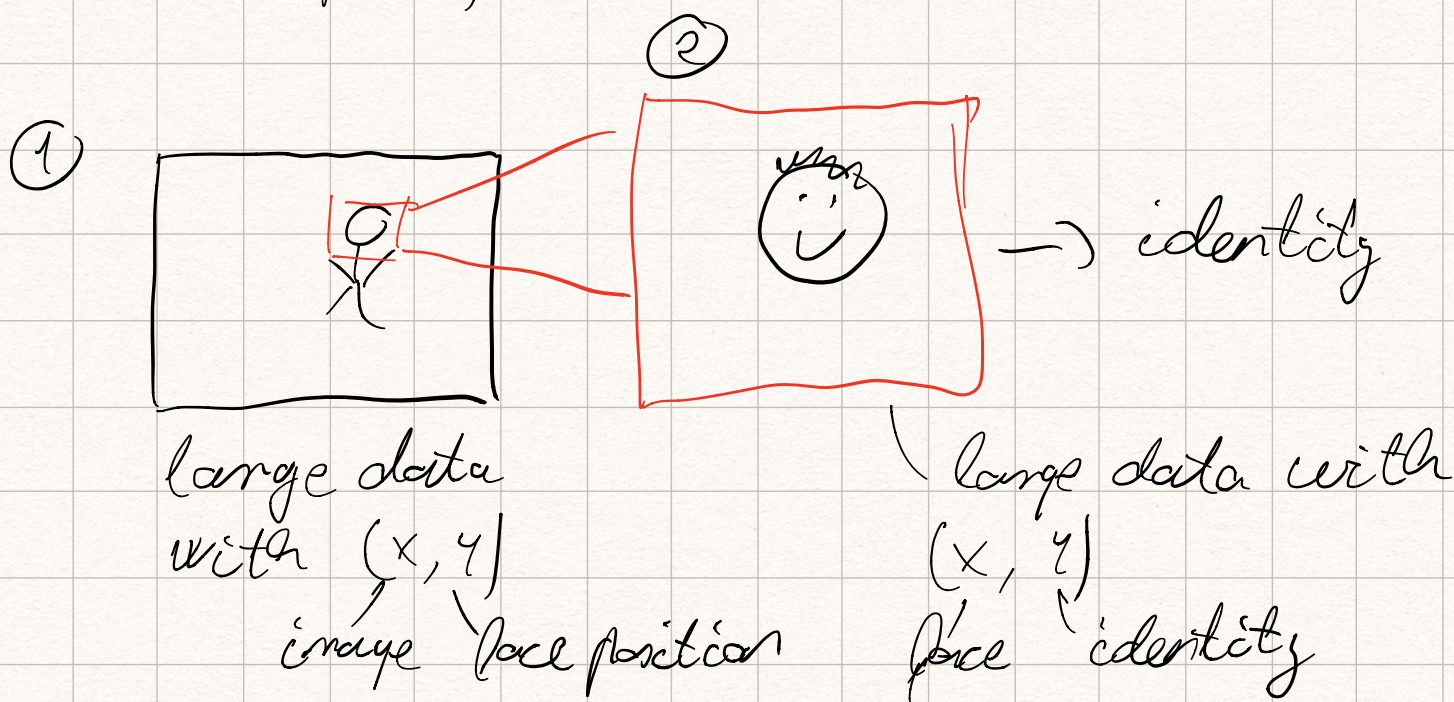
- shared lower level features
- amount of data for each label is similar
- You need big neural network to train well

End-to-end learning

Example face recognition:



to solve end-to-end we need too large dataset



But for example machine translation:

there is so many translation between english and french \Rightarrow end-to-end learning works well

Using a network for all Task: (end-to-end):

- image littering
- features extracting
- learning

Pros:

- less hand designed of components needed

Cons:

- need large amount of data
- excludes potentially useful hand-designed components