Conv net
- Multiple conv layers followed by a pool layer
- FC layers in the last few layers
- flatten volume before FC layers

LeNet-5 : deeper → # of channels ↑, width & height ↓

padding isn't used

ResNet : ResNet block can easily learn to approximate the identity ftn

→ not affect the performance of the network at all in worst case scenario

- data augmentation → computer vision system의 performance 개선
  - ↳ input: pixels (복잡한 구조를 지님)
  - 문제점: data가 충분하지 않음
- transfer learning, pre-trained model 등에 적용

> common methods

① mirroring

- 이미지 뒤집음 → preserve 전제 (같은 y값)

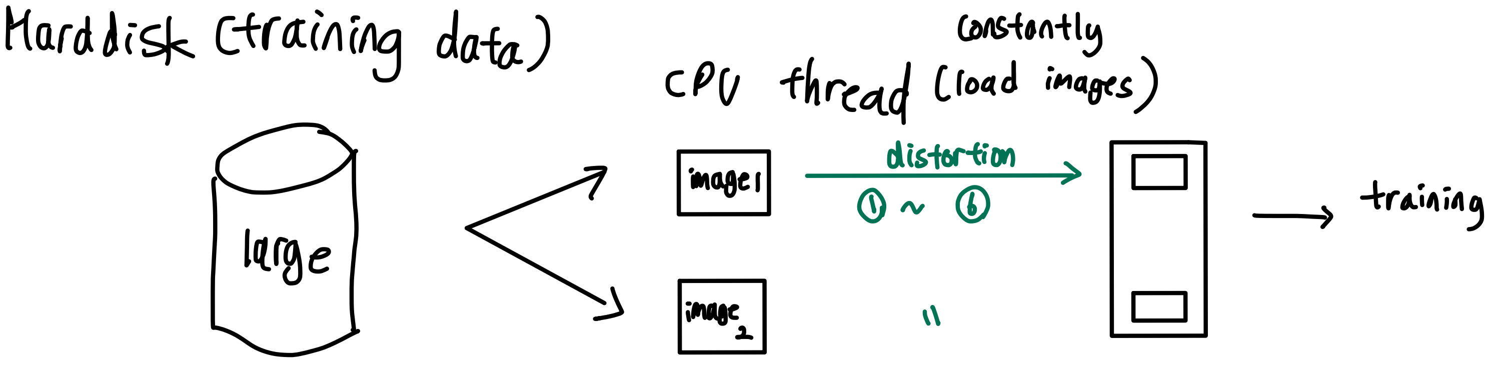
② random cropping

- 이미지 일부만 가져옴
- random crops should be reasonably large subsets of actual image

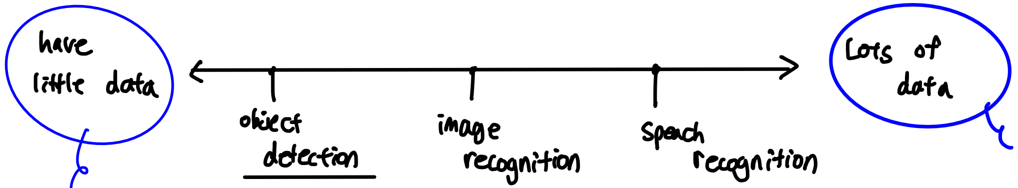
③ rotation
④ shearing
⑤ Local warping
} 실제로는 복잡성 때문에 덜 쓰임
①, ②가 주로 많이 쓰임.

⑥ Color shifting

- different color channels에 distortion 적용 ex) R G B
  +20 , -20 , +20
  - → more robust to changes in the colors of the image



one ↑ threads that load data / distortion and pass it to another thread that implements training



bounding box 까지 있어 비용↑

For good perform,
more hand-engineering
transfer learning ↑

Simpler algorithms
less hand-engineering (spend more time building up the learning system)

hand-engineering : hand design features, hand design network architectures → knowledge와 연관

Tips for (benchmarks / competitions !

① ensembling (average several (3~7) network outputs) → 1~2% better → 차이 several networks

② multi-crop at test time
: run classifier on multiple versions of test-images and average results → 1 network

But don't recommed applying them on production systems