

OpenCV for ML, NN, DL

21

OpenCV 클래스

□ cv2.ml

```
cv2.ml.ANN_MLP  
cv2.ml.DTrees  
cv2.ml.EM  
cv2.ml.KNearest  
cv2.ml.LogisticRegression  
cv2.ml.NormalBayesClassifier  
Cv2.ml.RTree  
cv2.ml.SVM  
cv2.ml.SVMSGD  
Cv2.ml_TrainData
```

□ cv2.dnn

```
cv2.dnn.blobFromImage  
cv.dnn.readNetFromTensorflow  
cv.dnn.readNetFromCaffe
```

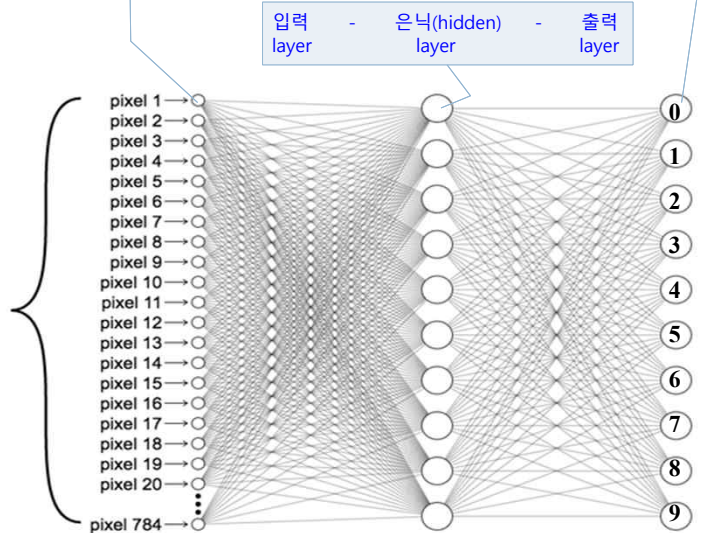
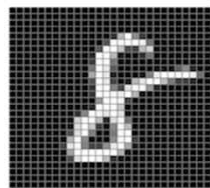
22

ANN(Artificial Neural Network)

10 개의 출력 뉴런

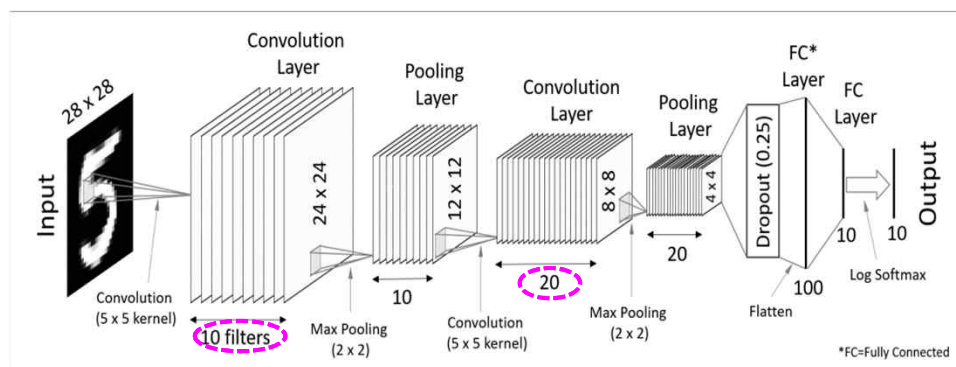
- MLP(Multi-Layer Perceptron)
 - 2 layer_BP(backpropagation)

입력 영상의 각 픽셀을 하나의 노드(뉴런)로 : 784 노드
입력층 내 뉴런 간 연결 없음. 각 뉴런은 독립적임



23

CNN



입력 영상에 다양한 convolution 마스크 적용
CNN에서 학습의 목표는 올바른 마스크를 찾는 것
학습데이터에 대해 올바른 마스크를 찾기 위해 계속 학습시킴.

24

예제 : MNIST 숫자 인식 with MLP/CNN

```
net = cv2.dnn.readNetFromTensorflow('./dnn/MNIST_MLP_frozen_graph.pb')
#net = cv2.dnn.readNetFromTensorflow('./dnn/MNIST_CNN_frozen_graph2.pb')

x_img = np.zeros(shape=(28, 28), dtype=np.uint8)
contours, _ = cv2.findContours(gray, mode, method)

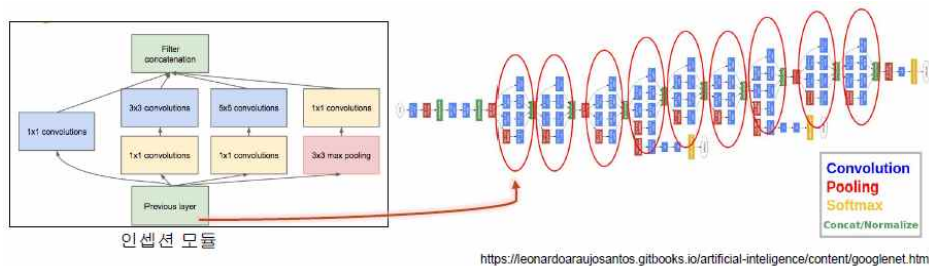
for i, cnt in enumerate(contours):
    ...
    cx, cy, r = int(cx), int(cy), int(r)
    img = gray[cy-r:cy+r, cx-r:cx+r]
    img = cv2.resize(img, dsize=(20, 20), interpolation=cv2.INTER_AREA)
    x_img[:, :] = 0
    x_img[4:24, 4:24] = img
    x_img = cv2.dilate(x_img, None, 2)
    x_img = cv2.erode(x_img, None, 4)

    blob = cv2.dnn.blobFromImage(x_img) # blob.shape=(1, 1, 28, 28)
    net.setInput(blob)
    res = net.forward()
y_predict = np.argmax(res, axis = 1) # 10개의 결과(숫자) 중 결과값이 가장 큰 숫자
```

25

CNN의 예

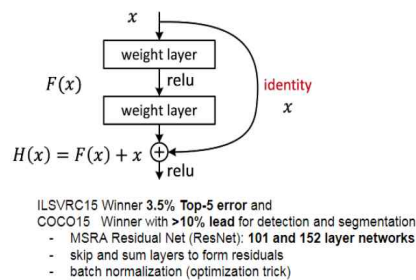
- **GoogleNet** : 더 깊게. 더 깊게. 하지만 너무 무겁지 않게.
 - ▣ 22층짜리 컨볼루션 신경망으로 **인셉션(Inception) 모듈**이라는 빌딩 블록을 반복적으로 사용
 - 인셉션 모듈(inception module)은 CNN의 최적인 지역적 희소 구조를 찾아내고 이를 사용 가능한 자원으로 근사함
 - ▣ ILSVRC 2014 winner (6.67% top 5 error)



26

CNN의 예

- **Deep Residual Networks (ResNet)** : 더 깊게, 그러나 더 빠르게
 - Microsoft 팀이 개발
 - **152층**짜리 컨볼루션 신경망
 - 층이 지나치게 깊으면 학습이 잘되지 않고 오히려 성능이 떨어질 수 있음
 - 해결책 : 층의 깊이에 비례해 성능을 향상시키기 위해 **스킵(Skip) 연결** 도입
 - ILSVRC 2015 winner (3.6% top 5 error)



27

예제 : 영상 분류(cnn)

with : 파이썬 2.5에서 도입된 기능
try/finally을 대신하여 더 간편하고 쉽게 사용
file을 열고, 사용한 후에 닫는 것을 보장

n01440764 tennis, Tinca tinca
n01443537 goldfish, Carassius auratus

```
with open('../data/synset_words.txt') as f:
    class_names = [' '.join(l.split(' ')[1:]).rstrip()
                    for l in f.readlines()]
# f(synset_words.txt)의 각 줄을 읽어서 (for l in f.readlines())
# 각 줄을 ' '로 분리한 후 첫 단어를 제외하고 (join(l.split(' ')[1:]))
# ' '와 2번째 이후 단어를 연결하고 오른쪽 공백 제거 (' '.join().rstrip())
# class_names에 배열로 저장 (class_names = [])

# ① GoogleNet(Inception) from Caffe
googlenet_caffe = cv2.dnn.readNetFromCaffe('../data/bvlc_googlenet.prototxt',
                                            '../data/bvlc_googlenet.caffemodel')
classify('shuttle.mp4', googlenet_caffe, 'data', 'prob', (104, 117, 123), class_names)

# ② ResNet-50 from Caffe
resnet_caffe = cv2.dnn.readNetFromCaffe('../data/resnet_50.prototxt',
                                          '../data/resnet_50.caffemodel')
mean = np.load('../data/resnet_50_mean.npy')
classify('shuttle.mp4', resnet_caffe, 'data', 'prob', mean, class_names)
```

28

예제 : 영상 분류(cnn)

isinstance(인스턴스, 클래스)

입력으로 받은 인스턴스가 그 클래스의 인스턴스인지를 판단. true/false 리턴

```
def classify(video_src, net, in_layer, out_layer, mean_val, category_names,
            swap_channels=False):
```

```
    if isinstance(mean_val, np.ndarray):
```

```
        tensor = cv2.dnn.blobFromImage(frame, 1.0, (224, 224), 1.0, False);
```

```
        tensor -= mean_val
```

```
    else:
```

```
        tensor = cv2.dnn.blobFromImage(frame, 1.0, (224, 224),
                                         mean_val, swap_channels);
```

blobFromImage(입력 영상,
scaleFactor, Size, mean, SwarR&B)

```
    net.setInput(tensor, in_layer);
```

```
    prob = net.forward(out_layer);
```

```
    prob = prob.flatten()
```

```
    r = 1
```

```
    for i in np.argsort(prob)[-5:]: # prob 중에서 값이 큰 5개에 대해
```

```
        txt = '"%s"; probability: %.2f' % (category_names[i], prob[i])
```

```
        cv2.putText(frame, txt, (0, frame.shape[0] - r*40),
```

```
                      cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2);
```

```
        r += 1
```

argsort(데이터)

데이터를 실제 정렬하는 것이 아니라
정렬 시 오름차순 순서만 알려줌

29

예제 : 영상 분류(cnn)

```
img = cv2.resize(frame, dsize=(int(frame.shape[1]/2), int(frame.shape[0]/2)))
```

```
r = 1
```

```
for i in np.argsort(prob)[-5:]:
```

```
    txt = '"%s"; probability: %.2f' % (category_names[i], prob[i])
```

```
    cv2.putText(img, txt, (0, img.shape[0] - r*40),
```

```
                  cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2);
```

```
    r += 1
```

```
cv2.imshow('classification', img)
```

30