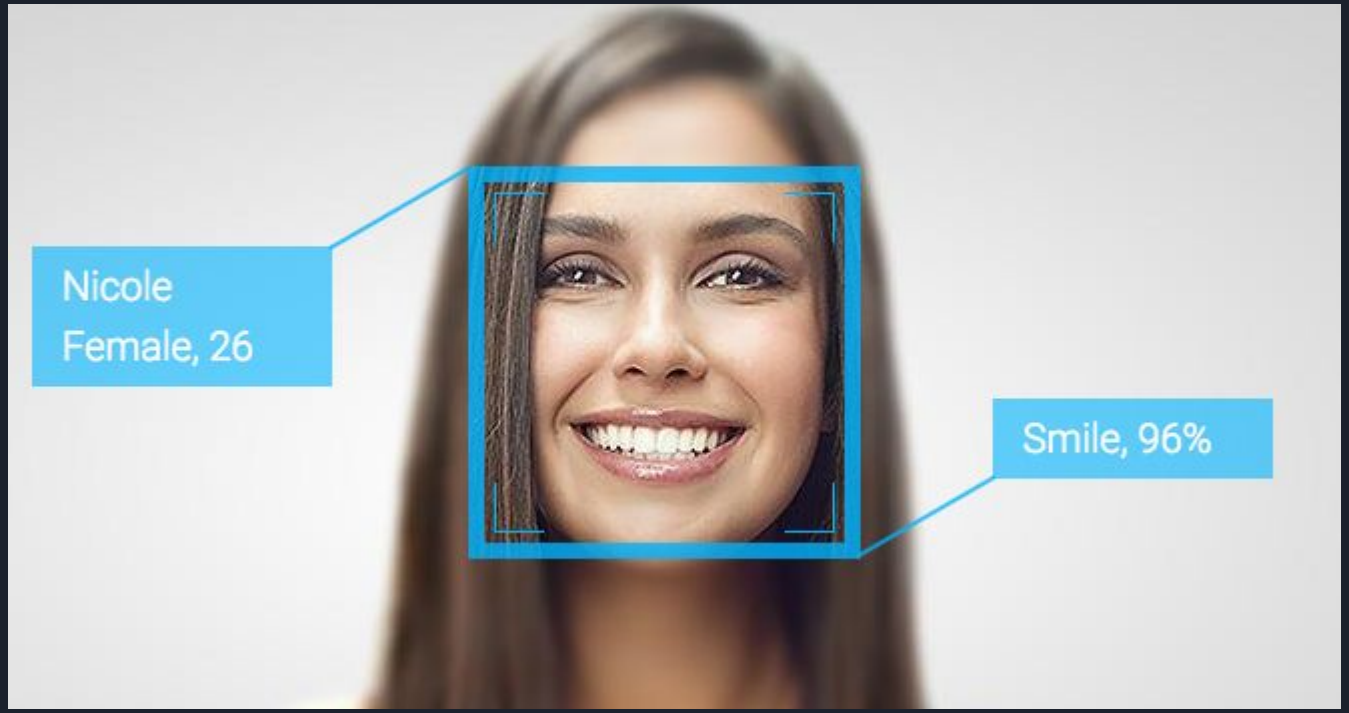
Emotion Recognition

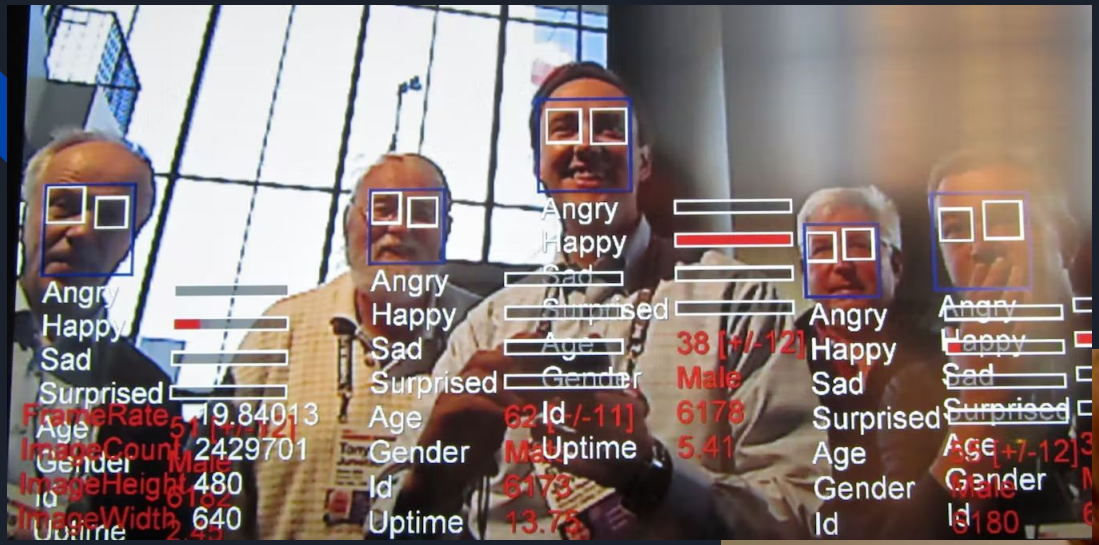
Spencer Goldberg

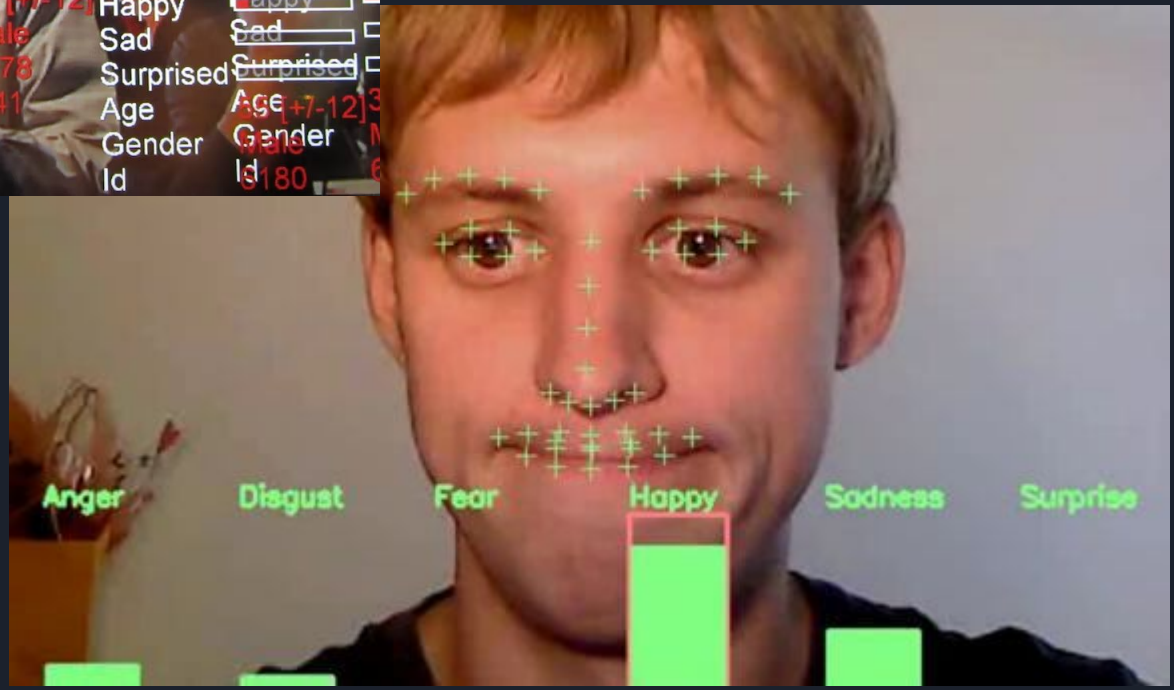
What are reasons to use emotion recognition?

Why Emotion Recognition in Faces?



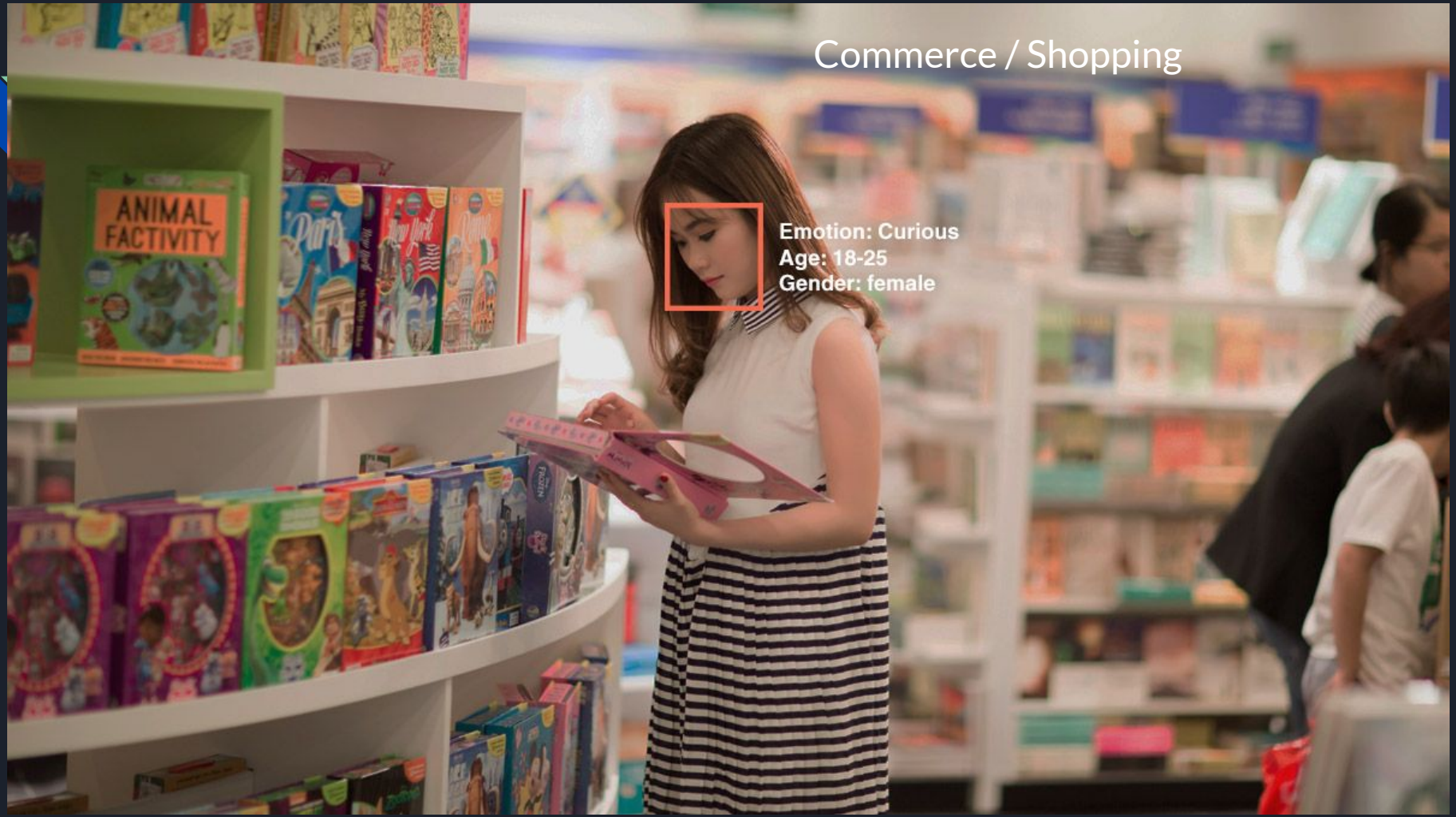
Product Sentiment



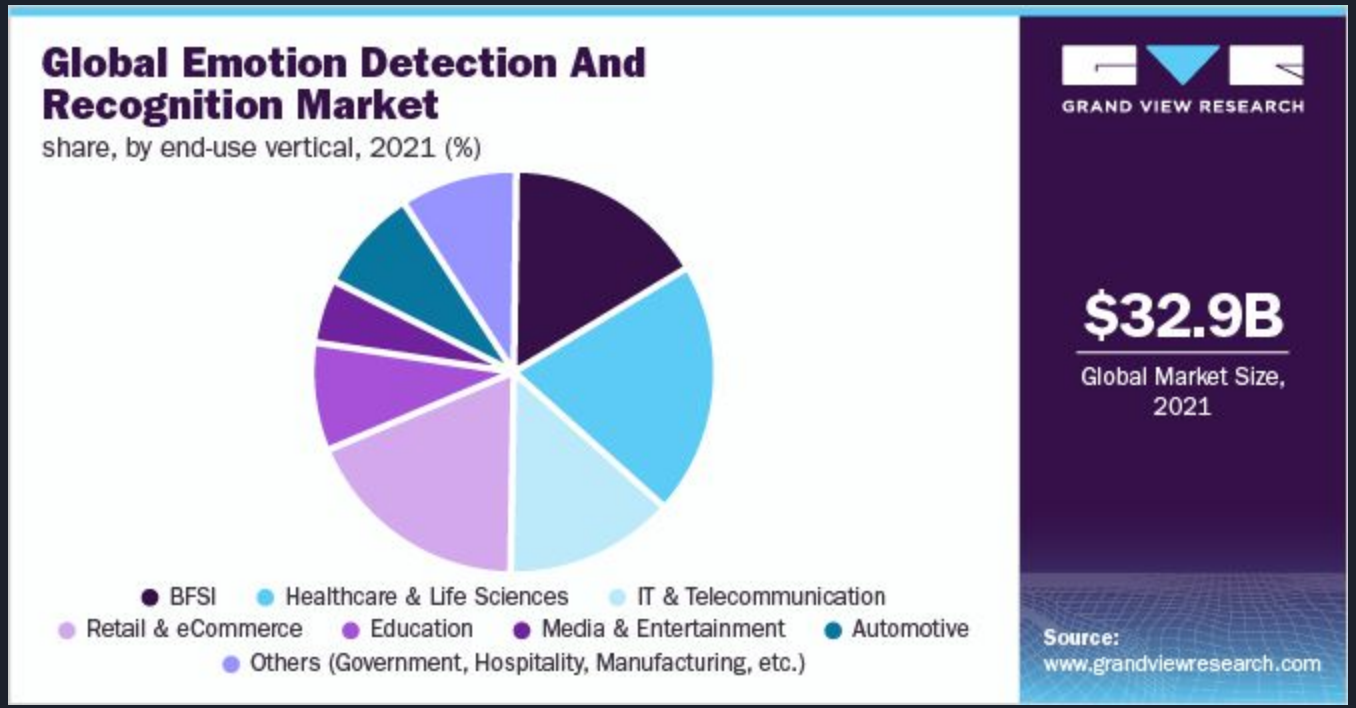


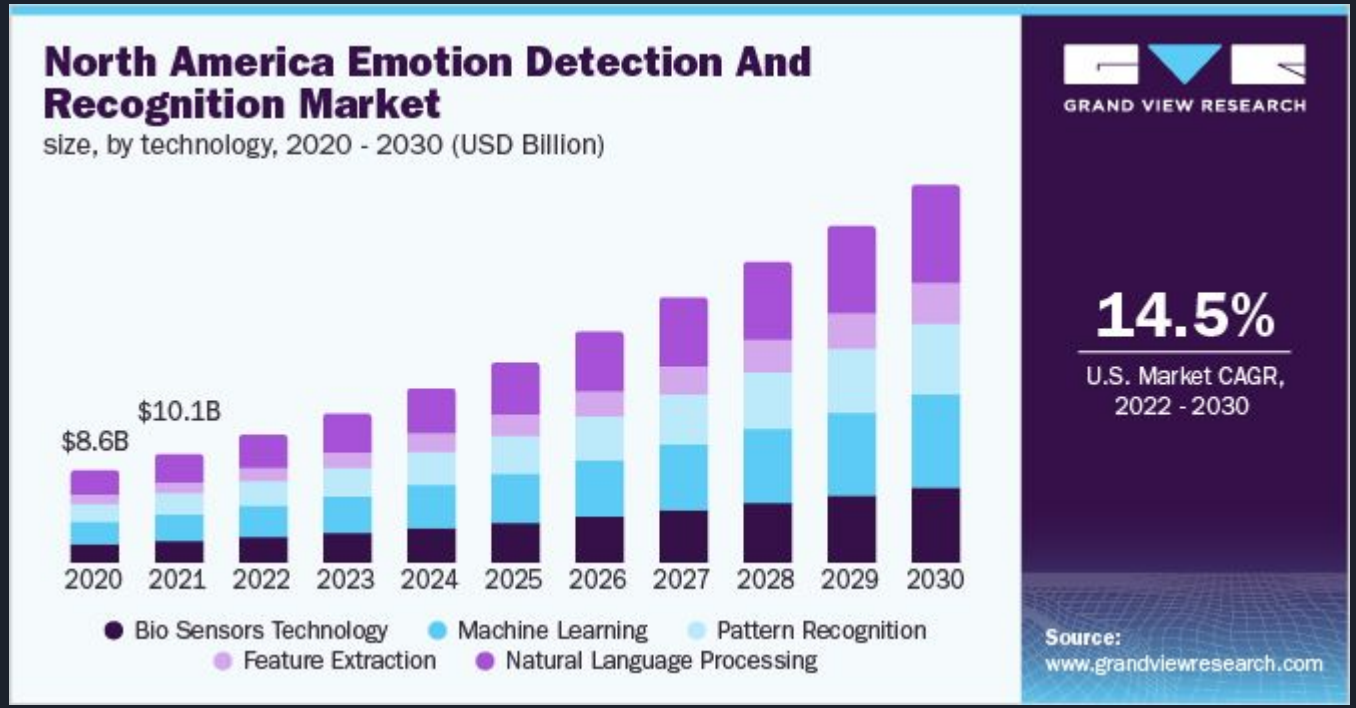
Marketing / Advertisements

Commerce / Shopping

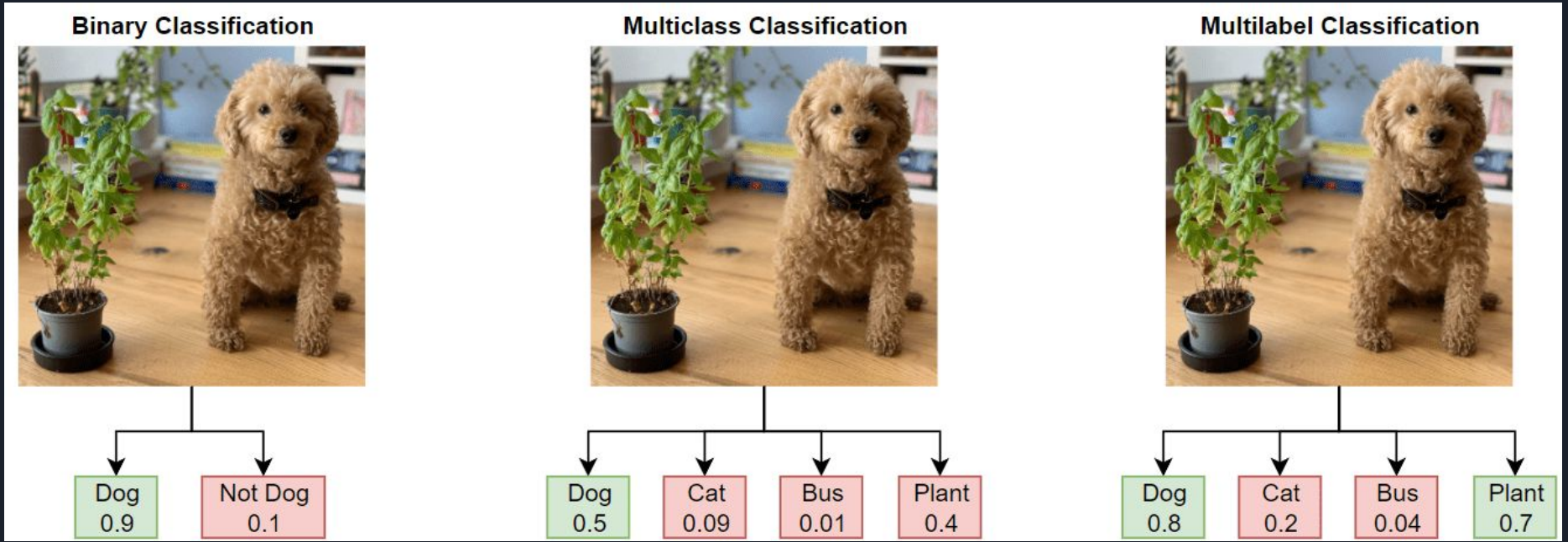


Market for Emotion Recognition

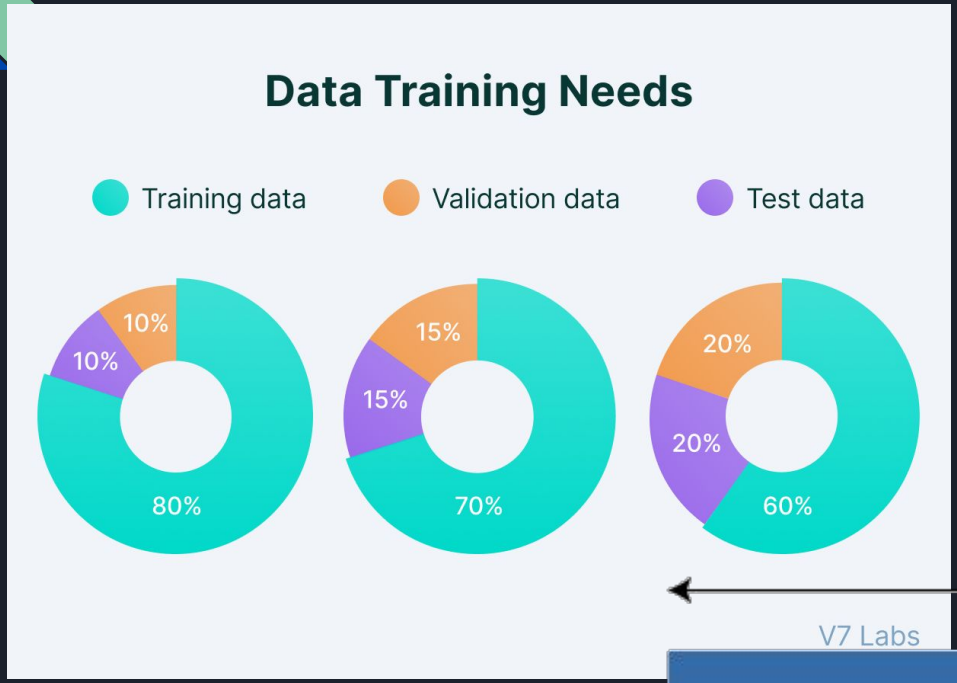


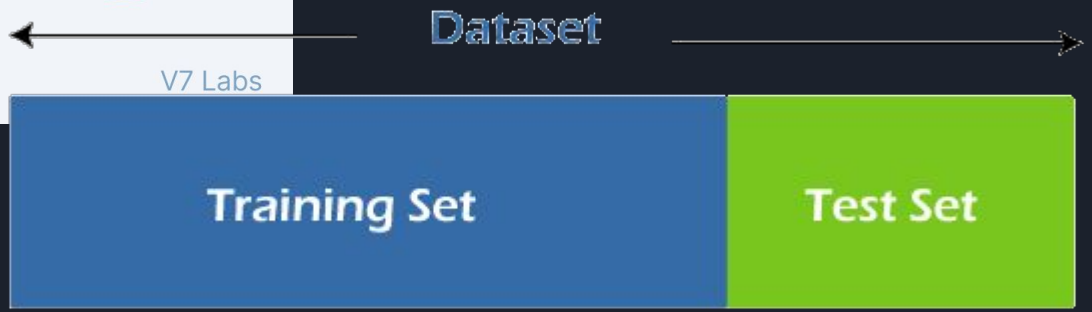

North America Emotion Detection And Recognition Market
size, by technology, 2020 - 2030 (USD Billion)
$8.6B
$10.1B
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
Bio Sensors Technology
Machine Learning
Pattern Recognition
Feature Extraction
Natural Language Processing
GRAND VIEW RESEARCH
14.5%
U.S. Market CAGR, 2022 - 2030
Source:
www.grandviewresearch.com

What is Image Classification?

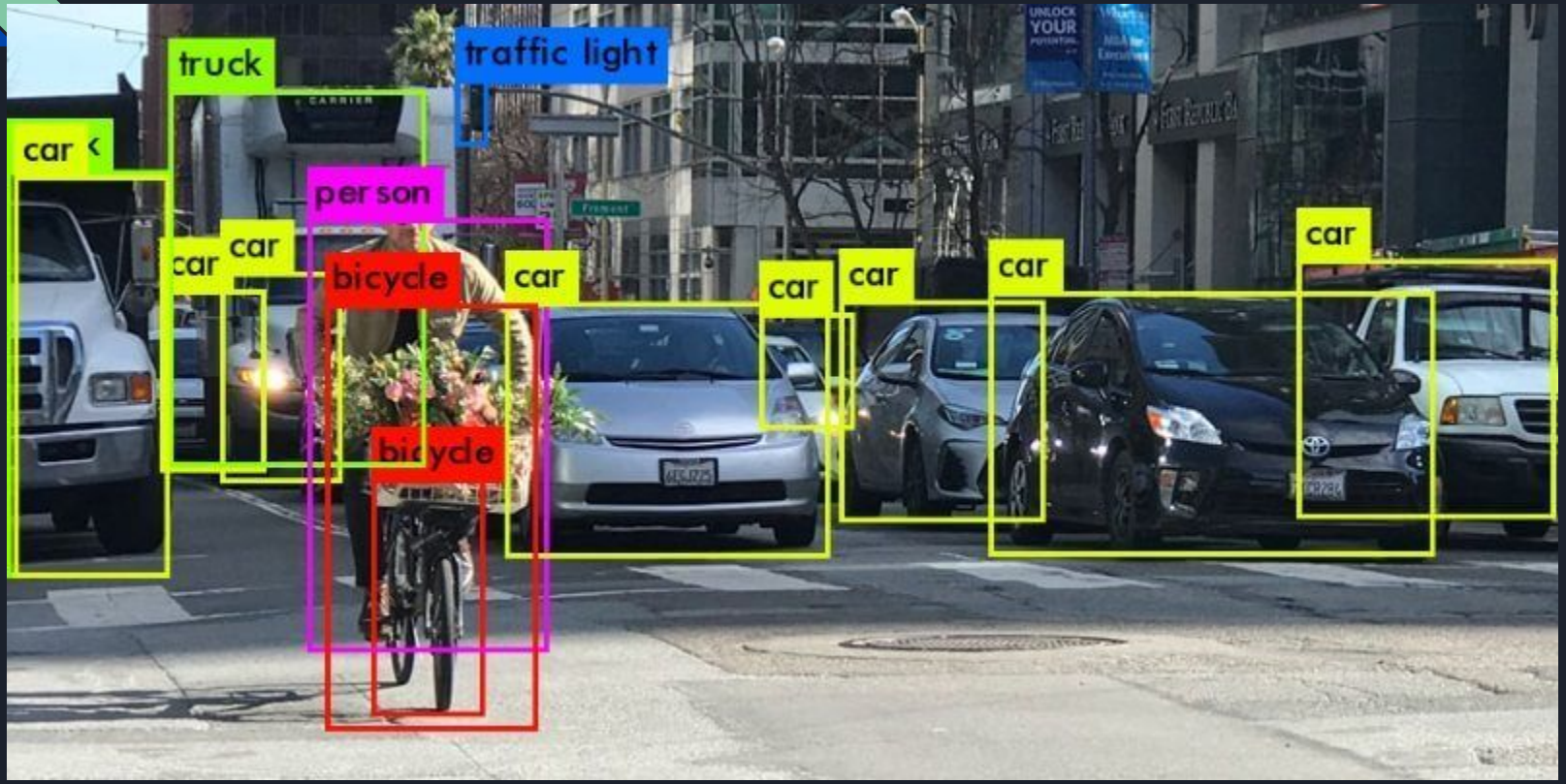


Splitting the data

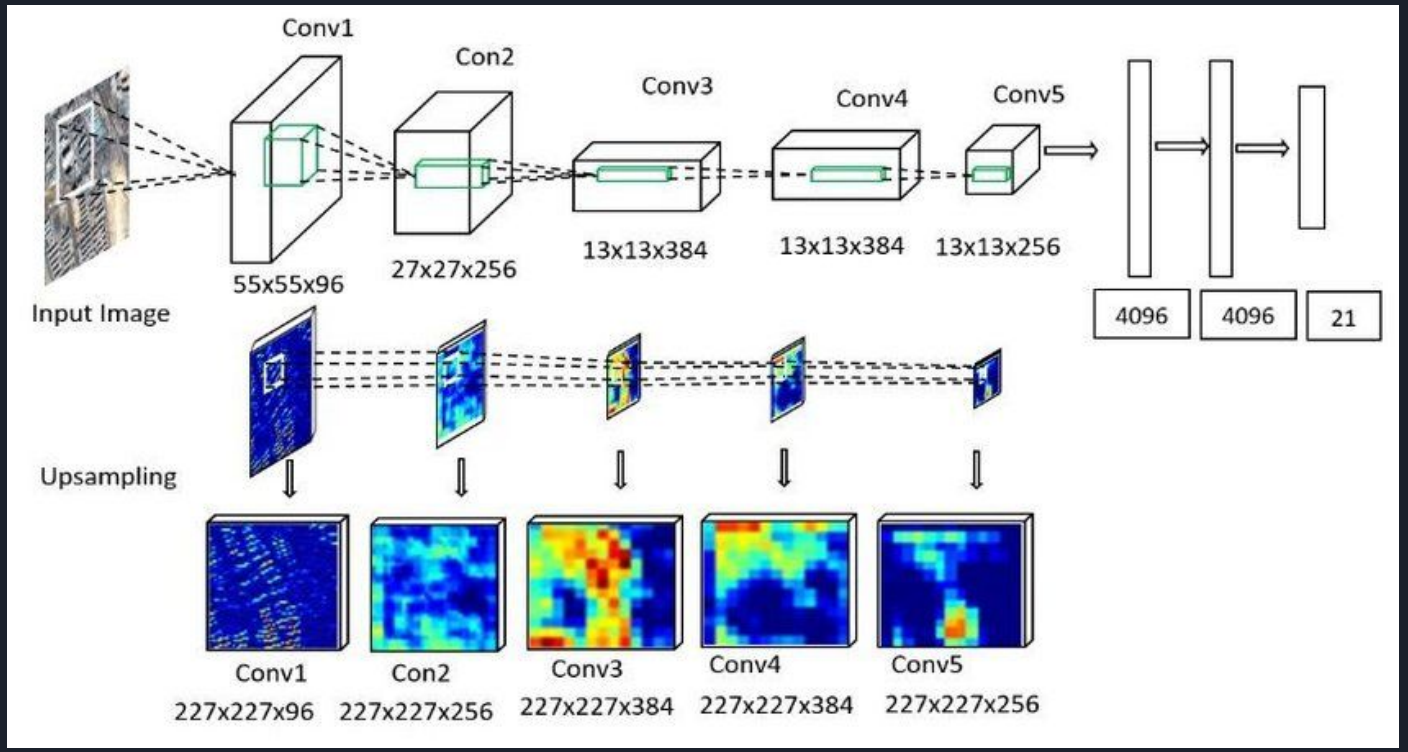


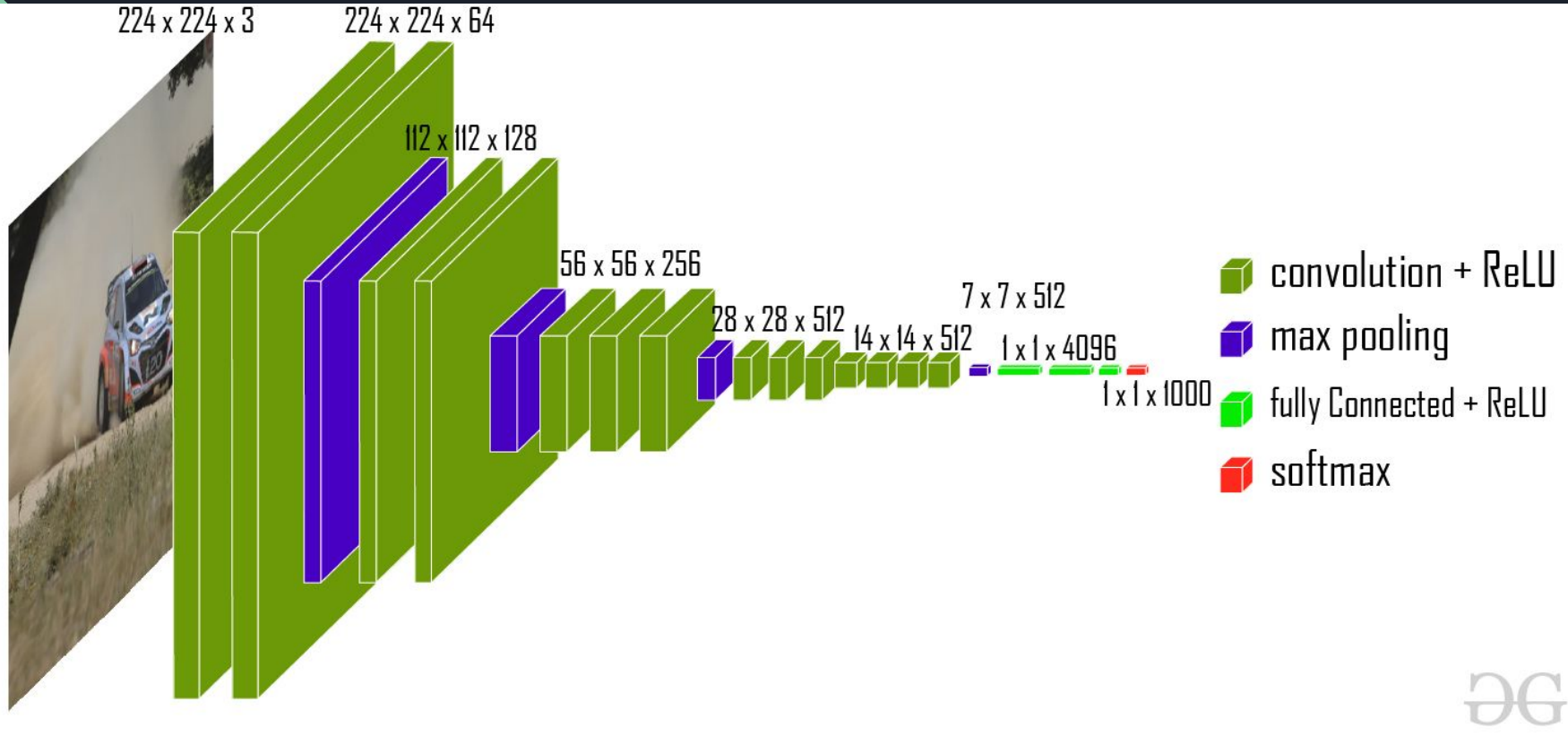


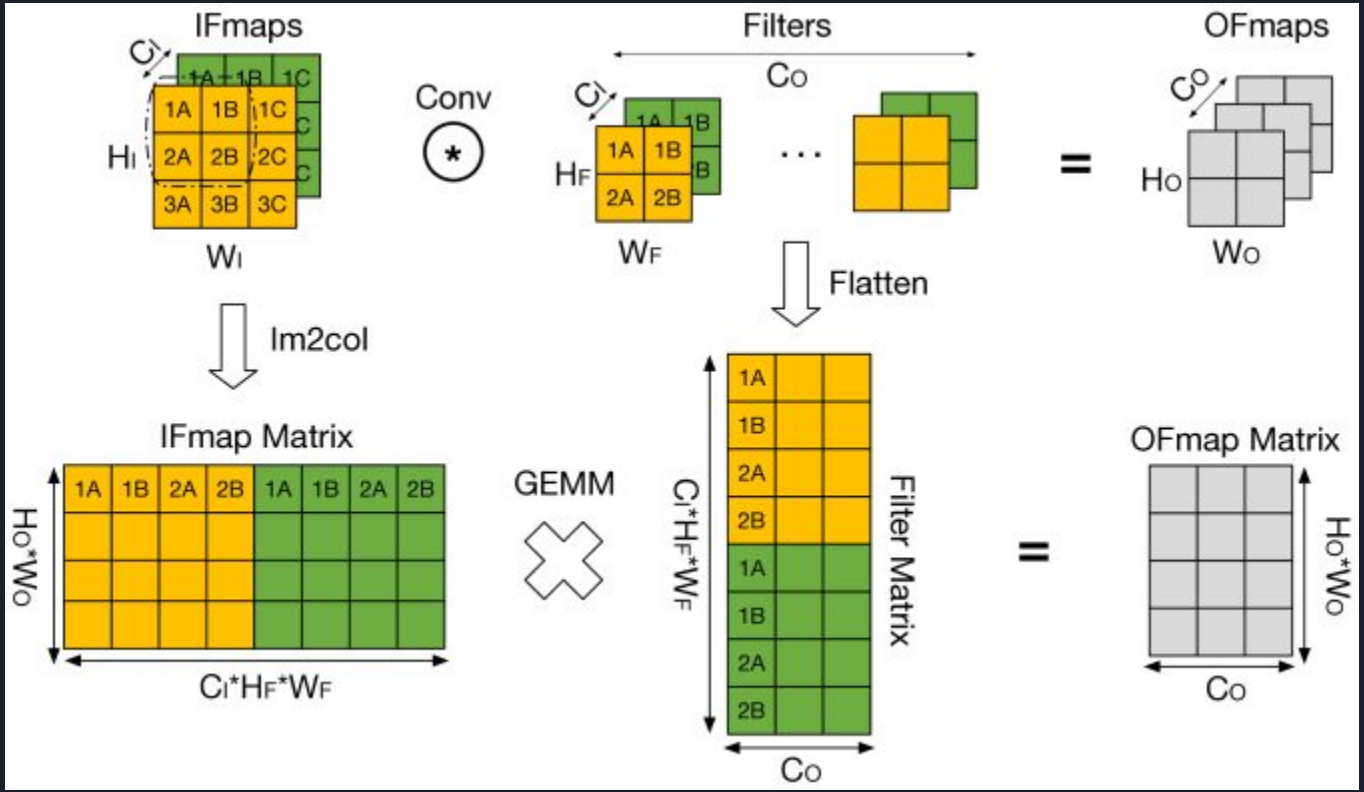
Computer Vision - Class Detection



AlexNet for Image Classification




224 x 224 x 3
224 x 224 x 64
112 x 112 x 128
56 x 56 x 256
28 x 28 x 512
14 x 14 x 512
7 x 7 x 512
1 x 1 x 4096
1 x 1 x 1000
convolution + ReLU
max pooling
fully Connected + ReLU
softmax


IFmaps
Ci
1A 1B 1C
2A 2B 2C
3A 3B 3C
HI
WI
Conv
*
Filters
Co
Ci
HF
WF
...
=
OFmaps
Co
Ho
Wo
Flatten
Im2col
IFmap Matrix
Ho*Wo
Ci*HF*WF
GEMM
Ci*HF*WF
Filter Matrix
Co
=
OFmap Matrix
Ho*Wo
Co

daisy, 95.8 %

	Class name	Score %
1	daisy	95.8214
2	ant	2.3421
3	bee	0.5741
4	pot	0.3236
5	pinwheel	0.2810

Emotion Recognition

https://github.com/spencergoldberg1/Data-Science/tree/develop/AlexNet

Classes to Look At: “Happy” and “Sad”

```
# 3. Pre-process the data and create data loaders
data_transforms = {
    'train': transforms.Compose([
        transforms.RandomResizedCrop(224),
        transforms.RandomHorizontalFlip(),
        transforms.ToTensor(),
        transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
    ]),
    'valid': transforms.Compose([
        transforms.Resize(256),
        transforms.CenterCrop(224),
        transforms.ToTensor(),
        transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
    ]),
}
```

Training Model and Accuracy

```
# Train the AlexNet model
alexnet.train()

num_epochs = 20

for epoch in range(num_epochs):
    print(epoch, " of ", num_epochs - 1)
    print('-' * 10)

    running_corrects = 0

    for inputs, labels in dataloaders["train"]:
        inputs = inputs.to(device)
        labels = labels.to(device)



        outputs = alexnet(inputs)
        preds = torch.max(outputs, 1)[1]



        optimizer.zero_grad()
        loss = criterion(outputs, labels)
        loss.backward()
        optimizer.step()

        running_corrects += torch.sum(preds == labels.data)

    print('Train Acc: {:.4f}'.format(running_corrects / dataset_sizes["train"]))
```

```
# Evaluate the AlexNet model on Validation Data
alexnet.eval()

running_corrects = 0

for inputs, labels in dataloaders["valid"]:
    inputs = inputs.to(device)
    labels = labels.to(device)



    outputs = alexnet(inputs)
    preds = torch.max(outputs, 1)[1]



    running_corrects += torch.sum(preds == labels.data)

acc_valid = running_corrects / dataset_sizes["valid"]
print('Valid Acc: {:.4f}'.format(acc_valid))
if acc_valid > 0.99:
    print("Done!")
    break
```

Final Model Accuracy

```
Train Acc: 0.9357
Valid Acc: 0.6625
Training complete
```

Test Data

```
trained_model = load_model(model_path)
```

```
urls = ['https://cdn.tinybuddha.com/wp-content/uploads/2016/01/Happy-Guy.jpg',
        'https://personalexcellence.co/files/girl-smiling2.jpg',
        'https://images.ctfassets.net/zkw0qlnf0vqv/psycom_page_fid6015_asset_6012/520ec3ffba27a19963640bfe09d2bbb7/51079685-depressed-tee
        'https://media.premiumtimesng.com/wp-content/files/2022/07/Photo-of-sad-man-used-to-illustrate-the-story-Source-_-Abayomiajayi.co
        'https://assets.entrepreneur.com/content/3x2/2000/20180627194538-GettyImages-828514788.jpeg',
        'https://hips.hearstapps.com/menshealth-uk/main/thumbs/30587/Smilingman.jpg',
        'https://sa1s3optim.patientpop.com/assets/images/provider/photos/2197018.jpg'
        ]
```

```
for image_url in urls:
  image = io.imread(image_url)
  plot(image)
  prediction = predict_image_url(image_url, trained_model)
  print("The predicted class for the input image is:", prediction)
```

The predicted class for the input image is: happy

The predicted class for the input image is: happy

The predicted class for the input image is: sad

The predicted class for the input image is: sad

The predicted class for the input image is: happy

The predicted class for the input image is: happy

The predicted class for the input image is: sad