# Deep Office

## Office Activity Analysis via Visual Recognition

Luyao Zhou, Xueyang Han, Yujia Li

# Introduction

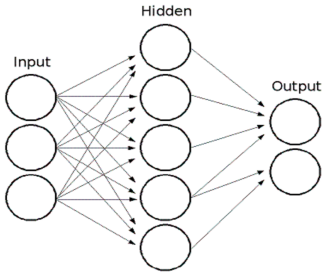
Daily working and studying: Efficiency VS Healthy

Activities:
Watching computer/Typing: **50%**
Writing on a notebook: **45%**
Standing/Moving around: **3%**
Exiting: **2%**
Using cell phone: **0%**
**Work too hard + not healthy**

Activities:
Watching computer/Typing: 10%
Writing on a notebook: 10%
Standing/Moving around: 40%
Exiting: 20%
Using cell phone: 20%

Work too less + not efficiency

# Introduction


Recorded Video

Continuous Images

CNNs

Timeline/Time Distribution

Simplify problem:

1) Considering 3 activities – sitting, standing, exiting;
2) Considering 1 target in one piece of video;

# Dataset Collection

- Train set includes images (320 x 240) of people sitting on an office chair, standing in an office, and people absence from an office (empty office).
- We collect all the images from 3 different sources: reusing parts of existing data set from Cornell Activity Dataset: CAD-60[4]; reusing parts of existing data set from LIRIS Human Activities Dataset[5]; collecting from recorded videos of team members.
- We decide to use 2 train set for training and predictions: the train set 1 has a total number of 10340 images, while the train set 2 has a total number of 13054 images. Train set 2 has about 26% more data comparing to train set 1.
- The test dataset has 8 pieces of videos of people(one person per video) doing sitting, standing and exiting activities, 5 minutes per video.

# Train Dataset

Count and Distribution of Train Set 1

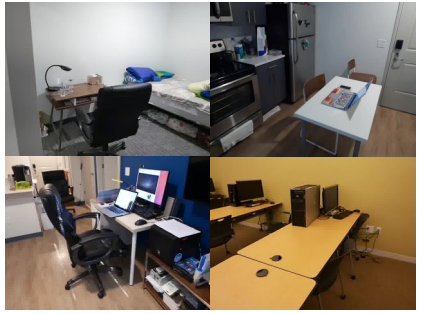
| Sources | Sit | Stand | Exit |
|---|---|---|---|
| CAD-60/LIRIS | 0 | 156 | 0 |
| Team Members | 3670 | 2754 | 3605 |
| Percentage | 36.0% | 28.6% | 35.4% |

Count and Distribution of Train Set 2

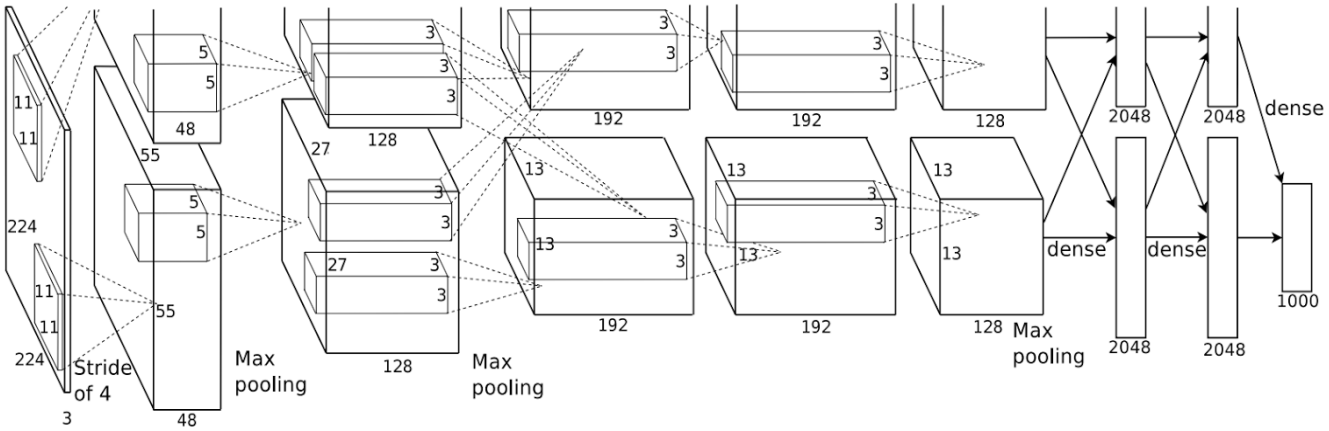
| Sources | Sit | Stand | Exit |
|---|---|---|---|
| CAD-60/LIRIS | 0 | 156 | 0 |
| Team Members | 4930 | 4364 | 3605 |
| Total | 37.8% | 34.6% | 27.6% |

- 3 People
- 7 Office Settings
- 13055 images
- Equally distributed





# Models: AlexNet




**Sequential 1**

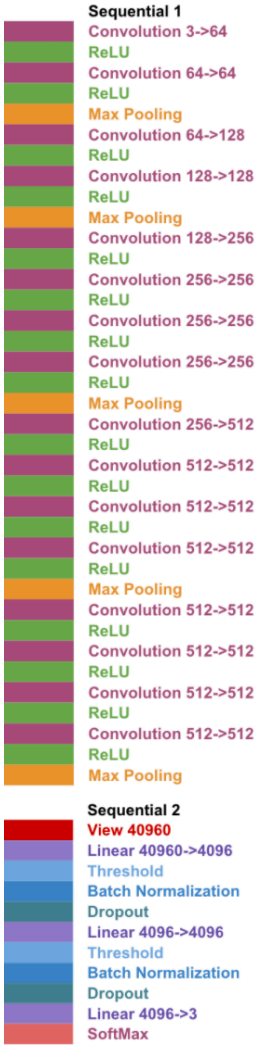
- Convolution 3->64
- Batch Normalization
- ReLU
- Max Pooling
- Convolution 64->192
- Batch Normalization
- ReLU
- Max Pooling
- Convolution 192->384
- Batch Normalization
- ReLU
- Max Pooling
- Convolution 384->256
- Batch Normalization
- ReLU
- Max Pooling
- Convolution 256->256
- Batch Normalization
- ReLU
- Max Pooling

**Sequential 2**

- View 12824
- Dropout
- Linear 12824->4096
- ReLU
- Dropout
- Linear 4096->4096
- ReLU
- Linear 4096->3
- SoftMax

# Models: VGG-19

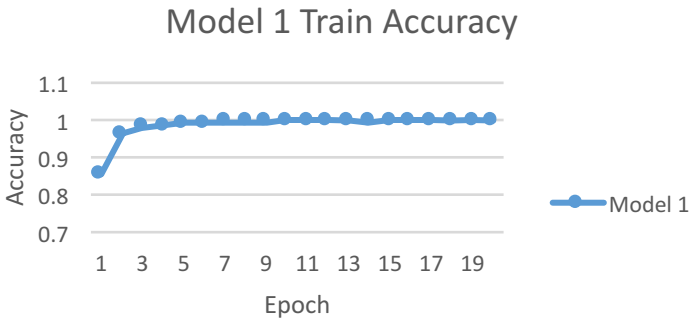
| ConvNet Configuration | | | | | |
|---|---|---|---|---|---|
| A | A-LRN | B | C | D | E |
| 11 weight layers | 11 weight layers | 13 weight layers | 16 weight layers | 16 weight layers | 19 weight layers |
| input (224 × 224 RGB image) | | | | | |
| conv3-64 | conv3-64 **LRN** | conv3-64 **conv3-64** | conv3-64 conv3-64 | conv3-64 conv3-64 | conv3-64 conv3-64 |
| maxpool | | | | | |
| conv3-128 | conv3-128 | conv3-128 **conv3-128** | conv3-128 conv3-128 | conv3-128 conv3-128 | conv3-128 conv3-128 |
| maxpool | | | | | |
| conv3-256 conv3-256 | conv3-256 conv3-256 | conv3-256 conv3-256 | conv3-256 conv3-256 **conv1-256** | conv3-256 conv3-256 **conv3-256** | conv3-256 conv3-256 **conv3-256** |
| maxpool | | | | | |
| conv3-512 conv3-512 | conv3-512 conv3-512 | conv3-512 conv3-512 | conv3-512 conv3-512 **conv1-512** | conv3-512 conv3-512 **conv3-512** | conv3-512 conv3-512 **conv3-512** |
| maxpool | | | | | |
| conv3-512 conv3-512 | conv3-512 conv3-512 | conv3-512 conv3-512 | conv3-512 conv3-512 **conv1-512** | conv3-512 conv3-512 **conv3-512** | conv3-512 conv3-512 **conv3-512** |
| maxpool | | | | | |
| FC-4096 | | | | | |
| FC-4096 | | | | | |
| FC-1000 | | | | | |
| soft-max | | | | | |

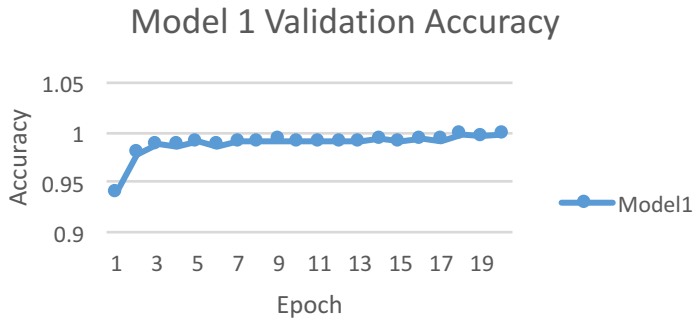


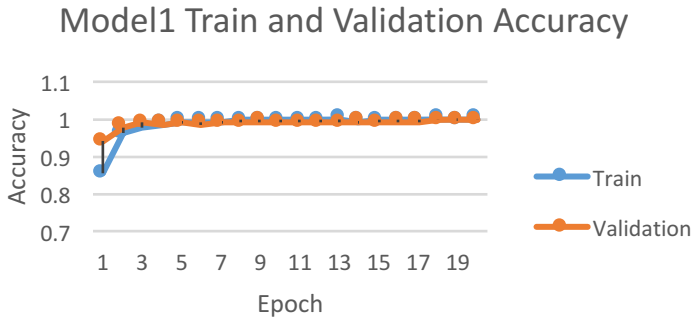
# Models

| Model | Based on | Pre-trained Weights | Trainset | Epoch |
|---|---|---|---|---|
| 1 | AlexNet | Yes | 1 | 20 |
| 2 | AlexNet | Yes | 2 | 20 |
| 3 | VGG-19 | Yes | 2 | 30 |

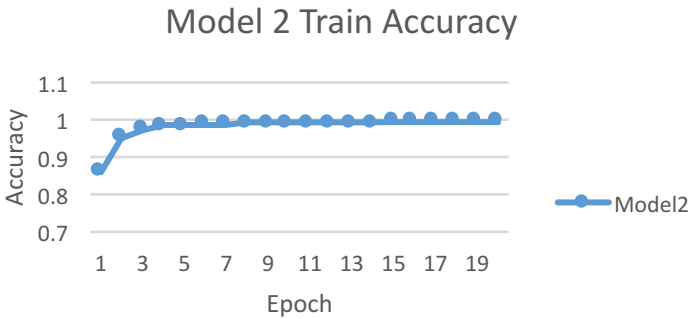
# Results: Model 1

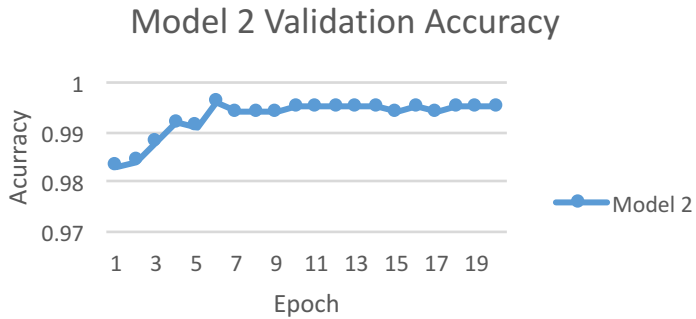


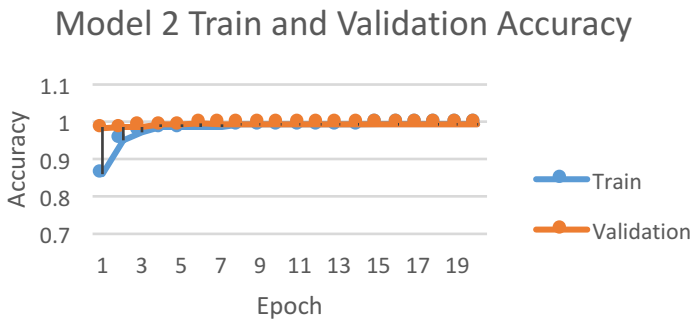



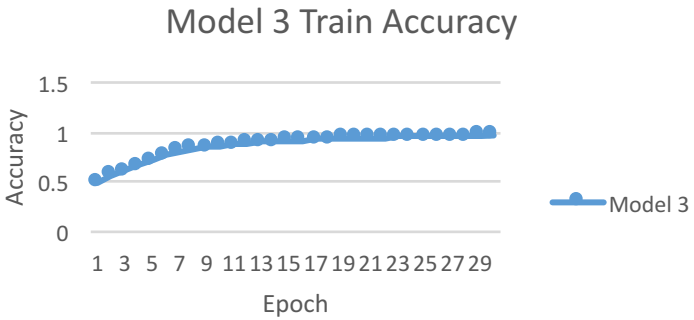
# Results: Model 2

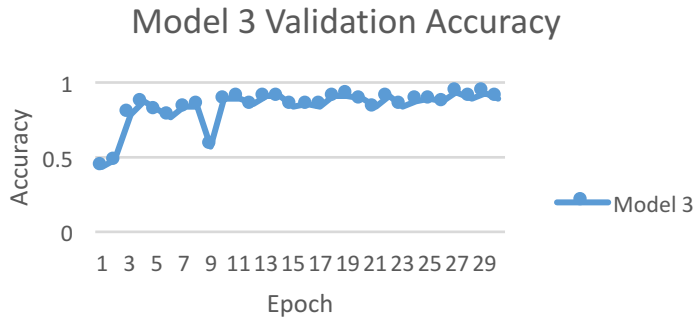


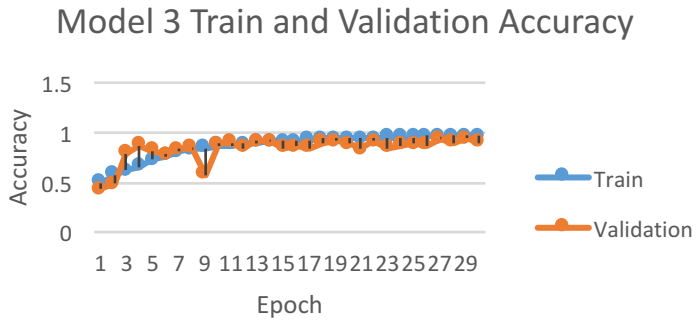



# Results: Model 3



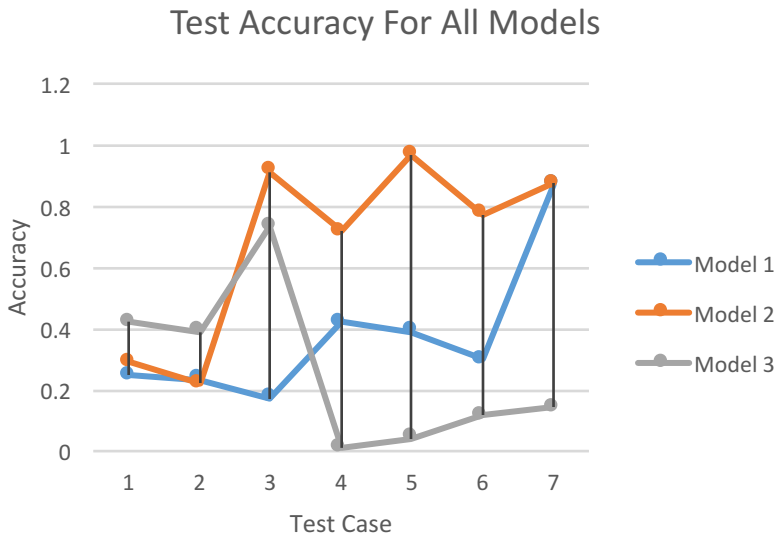



# Results

Test Accuracy for All Three Models

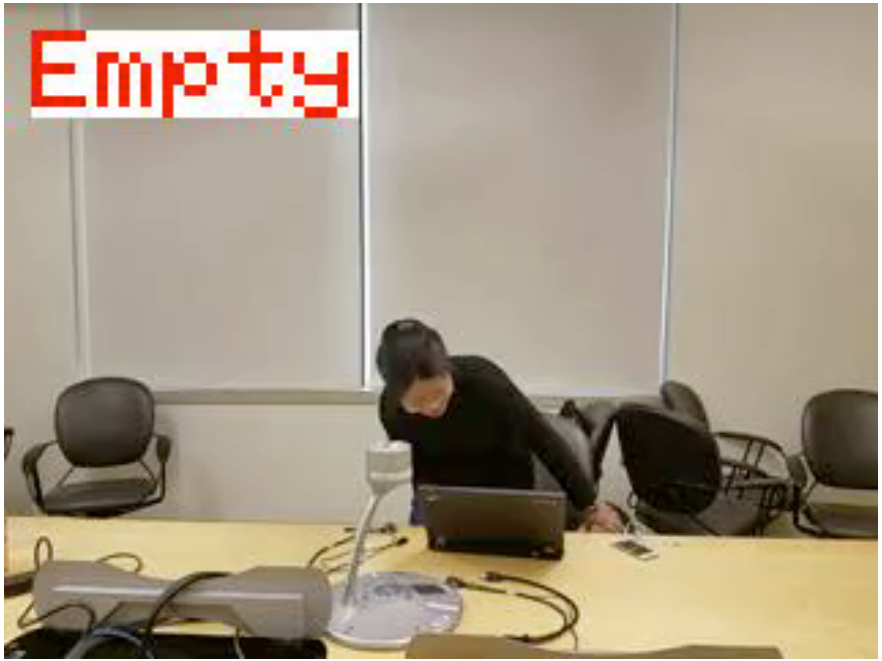
| Demo | Model 1 AlexNet Trainset 1 | Mode 2 AlexNet Trainset 2 | Model 3 VGG-19 Trainset 2 |
|---|---|---|---|
| Test 1 | 0.2463 | 0.2887 | 0.4200 |
| Test 2 | 0.2333 | 0.2207 | 0.3913 |
| Test 3 | 0.1713 | 0.9110 | 0.7340 |
| Test 4 | 0.4200 | 0.7167 | 0.0107 |
| Test 5 | 0.3900 | 0.9670 | 0.0407 |
| Test 6 | 0.2970 | 0.7723 | 0.1160 |
| Test 7 | 0.8690 | 0.8720 | 0.1410 |
| Average | 0.3753 | 0.6783 | 0.2648 |



# Demonstration

AlexNet
Output from Train set V1

Accuracy 42%

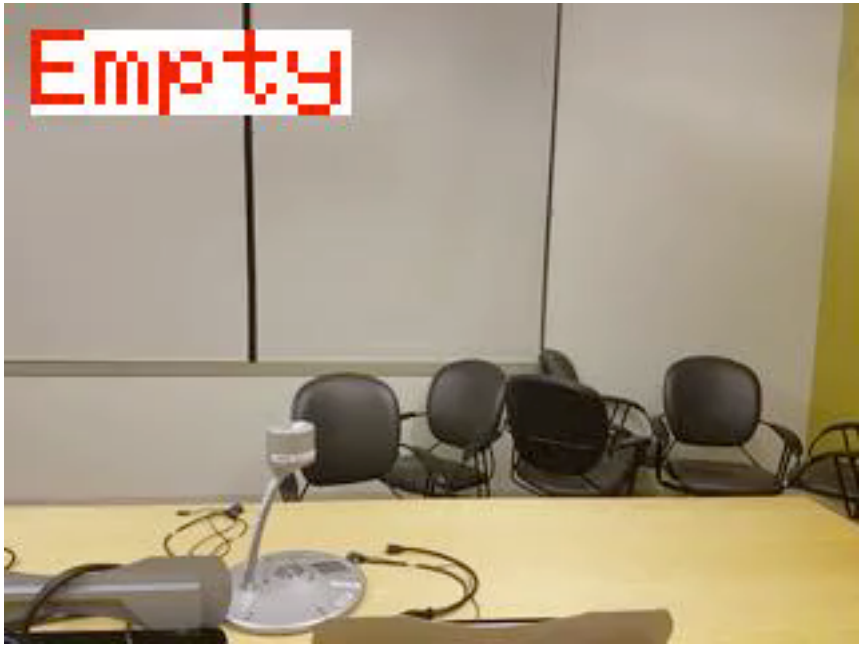


# Demonstration

AlexNet

Output from Train set V1

Accuracy 86.9%



# Closer Look


# Train Dataset

Count and Distribution of Train Set 1

| Sources | Sit | Stand | Exit |
|---|---|---|---|
| CAD-60/LIRIS | 0 | 156 | 0 |
| Team Members | 3670 | 2754 | 3605 |
| Percentage | 36.0% | 28.6% | 35.4% |

Count and Distribution of Train Set 2

| Sources | Sit | Stand | Exit |
|---|---|---|---|
| CAD-60/LIRIS | 0 | 156 | 0 |
| Team Members | 4930 | 4364 | 3605 |
| Total | 37.8% | 34.6% | 27.6% |

- 3 People
- 7 Office Settings
- 13055 images
- Equally distributed



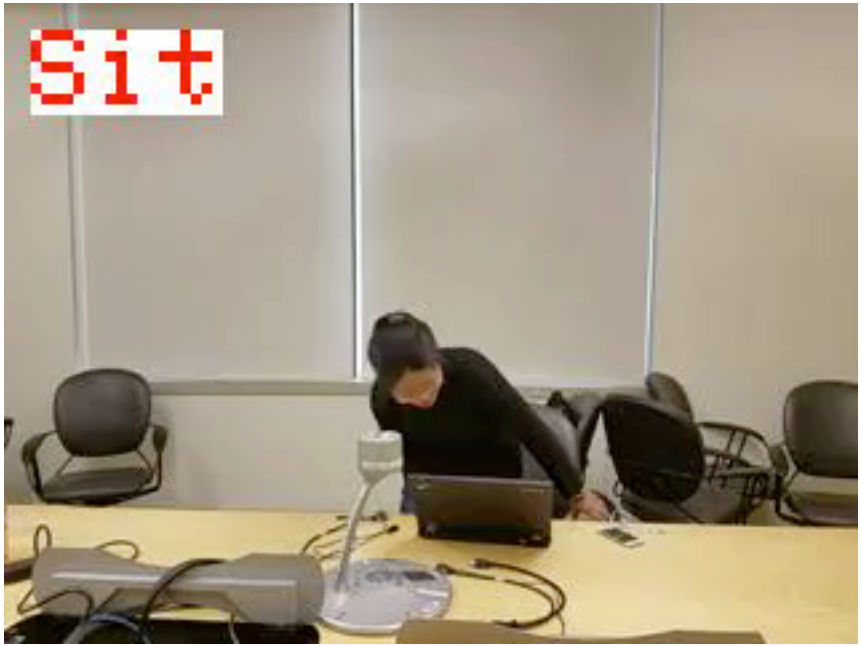


# Demonstration

AlexNet

Output from Train set V2

Accuracy improved from 42% to 86.9%



# Demonstration

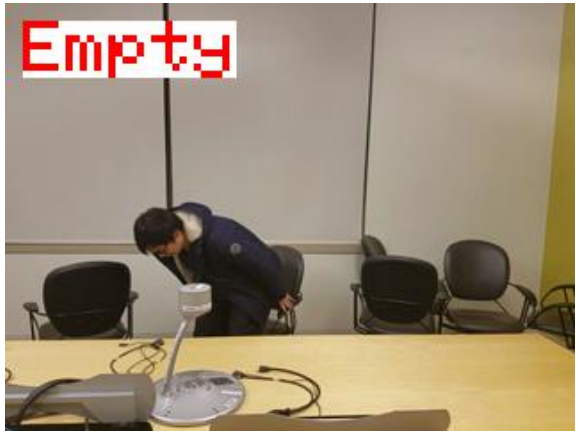
AlexNet

Output from Train set V2

Accuracy improved from 17.13%

to 91.1%

# Closer Look






# Discussion

**Trainset Size:** The test accuracy of model trained using train set 2 is higher comparing to model trained using train set 1

→ Train set with bigger size/more variety can help increase the test accuracy of this problem significantly.

**Models:** AlexNet model is giving much better results/higher test accuracy comparing to VGG model

→ Why?

**Test Video:** The consistence of the test video can dramatically affect the test result/test accuracy.

→ Example 1: sit -> stand -> empty -> stand -> sit –> stand -> empty

→ Example 2: sit -> stand -> empty

Questions?