

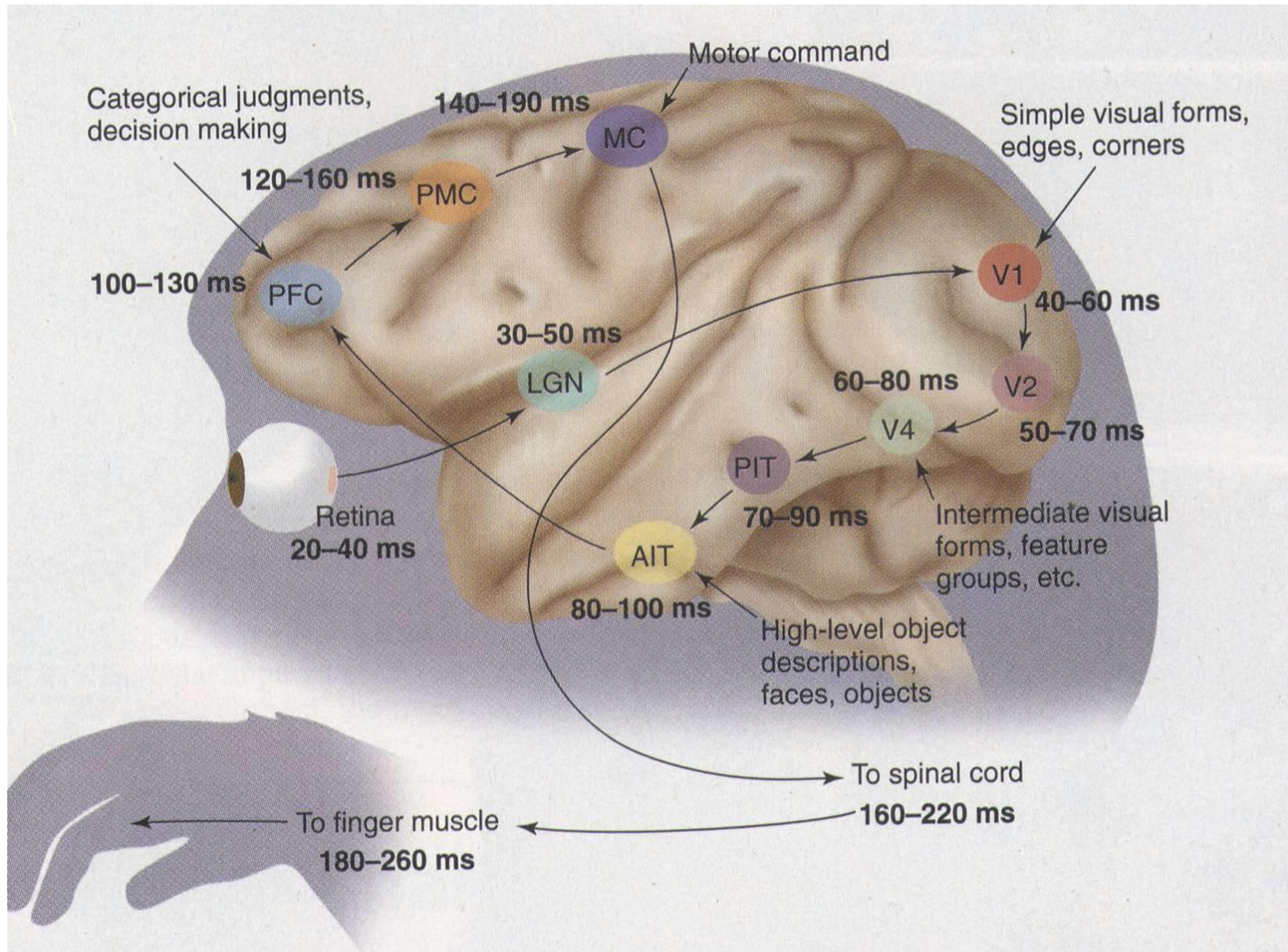
# Computer Vision:

## Making machines see

Roberto Cipolla  
Department of Engineering

<http://www.eng.cam.ac.uk/~cipolla/people.html>  
<http://www.toshiba.eu/eu/Cambridge-Research-Laboratory/>

# Vision: what is where by looking



# Computer Vision – What?

158 237 087 251 255 249 253 042 251 255 197 253 247 165 157 250 255 255 255 230 255 188 120 090 123 140 146 255 244 028 024 254 249 049 039 245 183 140 085 023 029 255 165 019 255 180 212 197  
236 245 109 066 190 251 255 252 088 253 255 144 125 075 245 255 255 254 255 246 194 045 092 181 096 245 086 211 234 031 016 180 255 047 029 188 236 254 120 041 032 226 193 043 203 246 255 255  
156 254 059 069 255 255 251 237 114 246 068 254 241 254 255 255 250 215 253 123 097 255 032 031 132 153 038 137 009 032 028 185 075 039 031 115 247 100 089 140 034 177 160 059 088 247 245 255  
088 222 132 249 245 181 195 231 244 095 252 037 255 253 255 255 255 251 248 253 255 095 061 072 224 026 169 153 034 027 036 202 043 035 144 182 216 137 158 018 254 155 046 119 194 117 255  
200 080 075 156 221 169 247 222 185 039 142 247 255 248 255 246 255 255 255 255 255 040 255 080 066 233 039 142 250 029 023 253 036 033 036 117 254 245 112 220 036 255 058 167 051 044 167 043  
145 085 062 200 035 239 225 255 246 028 042 248 255 253 065 253 255 239 247 186 151 056 252 155 137 184 255 210 255 043 025 255 130 031 041 137 255 163 189 225 022 223 031 091 107 033 096 238  
220 133 117 248 153 179 211 250 255 156 195 255 247 245 062 245 250 255 255 199 229 021 021 253 235 242 037 082 255 053 021 255 025 033 033 112 216 021 255 255 033 076 083 060 111 229 052 126  
244 223 157 142 207 168 224 253 255 157 255 191 246 249 114 249 251 255 255 255 236 076 031 030 255 218 198 192 255 057 022 244 022 032 032 052 222 028 219 253 035 157 159 027 061 202 102 098  
167 180 198 142 227 247 075 065 135 075 114 168 021 128 226 254 255 255 255 230 254 196 135 250 255 255 165 255 152 045 017 244 040 031 031 165 131 079 255 072 031 033 022 127 062 253 171 235  
071 253 236 124 255 063 083 084 059 032 069 054 087 064 147 069 230 255 249 254 243 253 029 255 190 177 219 186 116 210 022 255 037 025 015 255 025 230 255 247 029 149 020 046 069 192 239 108  
050 038 040 036 037 084 149 052 046 037 052 045 108 126 134 250 064 055 051 050 056 070 038 045 024 042 084 097 022 058 021 019 032 029 015 035 023 071 224 255 033 211 052 058 060 070 255 230  
044 056 046 032 042 091 170 084 095 035 062 049 097 135 137 056 052 034 047 041 053 037 029 026 043 025 037 044 045 055 017 017 028 026 032 042 015 020 084 231 031 021 030 029 076 145 213 254  
054 059 049 019 044 054 052 084 036 035 034 042 057 063 106 065 047 117 044 072 067 064 050 084 062 028 058 058 068 036 022 035 027 023 043 041 015 035 034 046 025 035 026 034 035 254 116 199  
029 070 048 029 030 051 084 041 076 032 042 074 097 092 089 083 068 068 072 068 042 029 053 060 029 061 070 049 043 027 032 028 026 030 039 023 041 044 034 032 038 035 044 039 145 226 237  
050 067 040 024 039 027 052 065 092 035 060 101 117 092 086 063 071 076 072 077 030 060 025 059 047 030 040 047 058 053 032 036 027 031 053 049 028 034 043 036 027 041 057 033 035 235 138 253  
055 065 050 030 081 032 057 068 068 033 060 079 098 088 084 087 044 036 068 068 065 032 061 045 061 024 039 045 043 088 031 035 029 035 040 034 057 030 055 024 027 026 036 035 025 198 242 025  
055 071 086 023 022 039 069 068 029 036 056 083 103 095 083 079 078 051 080 088 080 057 031 061 048 023 049 101 105 084 057 035 024 050 041 040 035 022 041 024 025 030 030 027 032 079 179 255  
053 074 049 030 042 049 076 088 037 031 051 076 084 125 077 096 094 068 069 048 086 040 057 057 067 016 046 050 048 072 056 031 020 037 055 040 045 023 027 020 033 029 047 031 045 147 237 255  
048 057 032 026 042 040 060 080 064 030 050 078 089 081 068 073 087 064 055 055 083 032 067 063 084 022 044 029 043 083 073 030 029 054 026 025 045 023 024 022 018 031 037 028 036 151 255 255  
056 052 067 026 093 051 081 065 029 033 039 049 095 128 052 090 034 046 045 049 069 022 055 028 034 023 036 032 042 084 038 026 028 038 037 060 039 028 022 023 025 021 025 031 068 081 255 255  
047 055 058 040 051 045 043 066 048 028 038 041 067 078 039 061 056 051 056 069 082 030 042 058 042 030 039 036 035 090 036 028 025 036 037 051 035 039 020 064 019 018 024 025 066 054 251 083  
053 055 053 027 054 050 047 042 041 033 038 038 044 046 046 047 052 061 041 087 044 024 047 041 037 024 035 044 033 081 038 026 024 036 021 043 027 026 023 047 025 015 026 023 062 101 107 077  
088 097 089 086 085 080 082 088 085 083 081 086 085 094 096 102 106 096 114 098 057 061 052 048 036 028 035 029 027 081 036 020 026 031 024 039 032 034 025 043 020 026 021 010 047 018 022 066  
101 104 097 100 104 092 093 094 093 088 037 079 078 079 079 085 076 080 080 082 081 101 099 083 098 094 034 043 028 074 051 015 029 104 102 086 022 040 025 028 022 027 080 092 069 048 127 135  
039 094 088 079 080 086 089 101 097 094 108 096 095 097 090 086 102 097 094 099 077 093 096 091 102 068 056 039 036 088 029 018 023 078 066 070 026 042 024 073 018 029 112 123 107 145 131 059  
079 089 083 079 088 086 077 078 081 081 020 082 074 075 073 077 073 072 099 072 016 096 097 096 081 061 032 034 039 109 027 012 024 097 115 143 034 049 028 101 064 077 129 136 118 113 087 035  
054 086 118 077 082 066 071 078 081 061 018 099 055 044 049 086 056 054 051 058 015 065 078 077 082 073 027 037 034 106 016 015 027 081 109 095 031 046 060 085 067 140 121 113 091 120 085 012  
052 088 060 062 048 063 053 045 040 031 013 022 045 023 054 094 030 048 032 040 017 062 082 071 068 068 023 031 032 112 028 021 025 118 033 057 068 102 117 117 101 093 090 082 109 062 067 059  
081 060 054 045 016 048 066 037 041 041 033 016 038 056 036 061 094 021 041 032 040 017 036 055 059 058 062 014 026 032 120 071 053 071 050 074 045 113 124 131 132 121 105 124 124 082 085 086 085  
043 078 022 036 038 047 071 045 039 043 021 040 041 046 044 099 134 047 043 015 110 038 059 047 075 080 021 071 069 071 075 106 102 110 127 130 131 129 120 124 119 111 072 083 080 090 093 090  
061 072 032 035 027 024 061 081 028 062 056 027 042 034 020 050 083 040 026 052 037 035 036 040 071 041 070 084 082 100 117 106 105 095 115 113 110 110 108 096 082 102 127 081 086 092 093 089 087  
025 077 033 028 019 023 057 035 016 050 025 058 025 029 043 081 043 080 067 077 094 102 102 115 098 102 099 096 096 088 097 101 099 095 103 090 092 113 084 059 095 089 086 088 091 091 088 085  
047 071 042 051 053 062 070 085 090 075 083 093 099 099 103 093 104 084 069 073 092 076 080 090 084 093 103 095 097 095 088 089 075 111 084 091 096 079 099 086 095 091 088 086 076 077 073 077  
088 090 094 070 100 084 072 078 081 099 106 101 096 087 086 082 089 085 066 067 083 079 081 073 088 092 085 074 068 049 070 065 055 060 091 112 063 082 073 076 075 083 088 095 091 094 095 095  
072 077 075 077 066 067 088 065 062 061 060 066 059 061 060 060 065 061 073 069 065 071 050 045 045 039 038 029 038 053 074 141 055 068 078 102 096 097 095 095 097 095 094 095 094 099 095 097

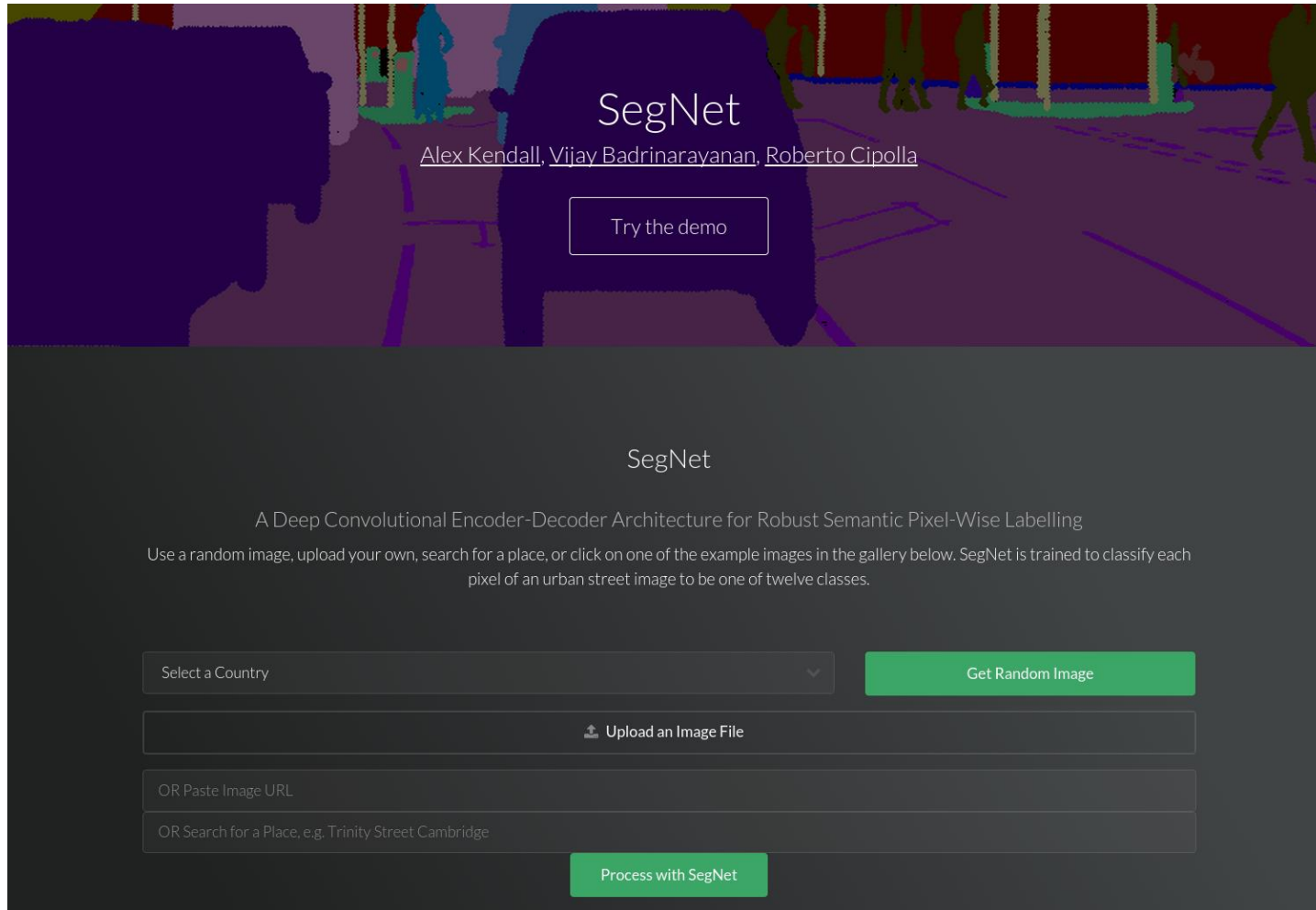


# Computer Vision – What?

---



# Real-time application



The image shows a web interface for the SegNet application. At the top, there is a header image of a street scene with a car and pedestrians, overlaid with a semi-transparent dark box containing the text 'SegNet' and the authors' names: 'Alex Kendall, Vijay Badrinarayanan, Roberto Cipolla'. Below this is a 'Try the demo' button. The main content area has a dark background and contains the following text: 'SegNet', 'A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling', and 'Use a random image, upload your own, search for a place, or click on one of the example images in the gallery below. SegNet is trained to classify each pixel of an urban street image to be one of twelve classes.' Below the text are four input fields: a dropdown menu for 'Select a Country', a green button for 'Get Random Image', an 'Upload an Image File' button with a file upload icon, and a text input field for 'OR Paste Image URL'. At the bottom, there is another text input field for 'OR Search for a Place, e.g. Trinity Street Cambridge' and a green button for 'Process with SegNet'.

SegNet

Alex Kendall, Vijay Badrinarayanan, Roberto Cipolla

Try the demo

SegNet

A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling

Use a random image, upload your own, search for a place, or click on one of the example images in the gallery below. SegNet is trained to classify each pixel of an urban street image to be one of twelve classes.

Select a Country

Get Random Image

Upload an Image File

OR Paste Image URL

OR Search for a Place, e.g. Trinity Street Cambridge

Process with SegNet

# Overview

---

1. Background: why and how?
2. 3R's of Computer Vision:
  - Reconstruction
  - Registration
  - Recognition

# 1. How to make machines that see?

# How?

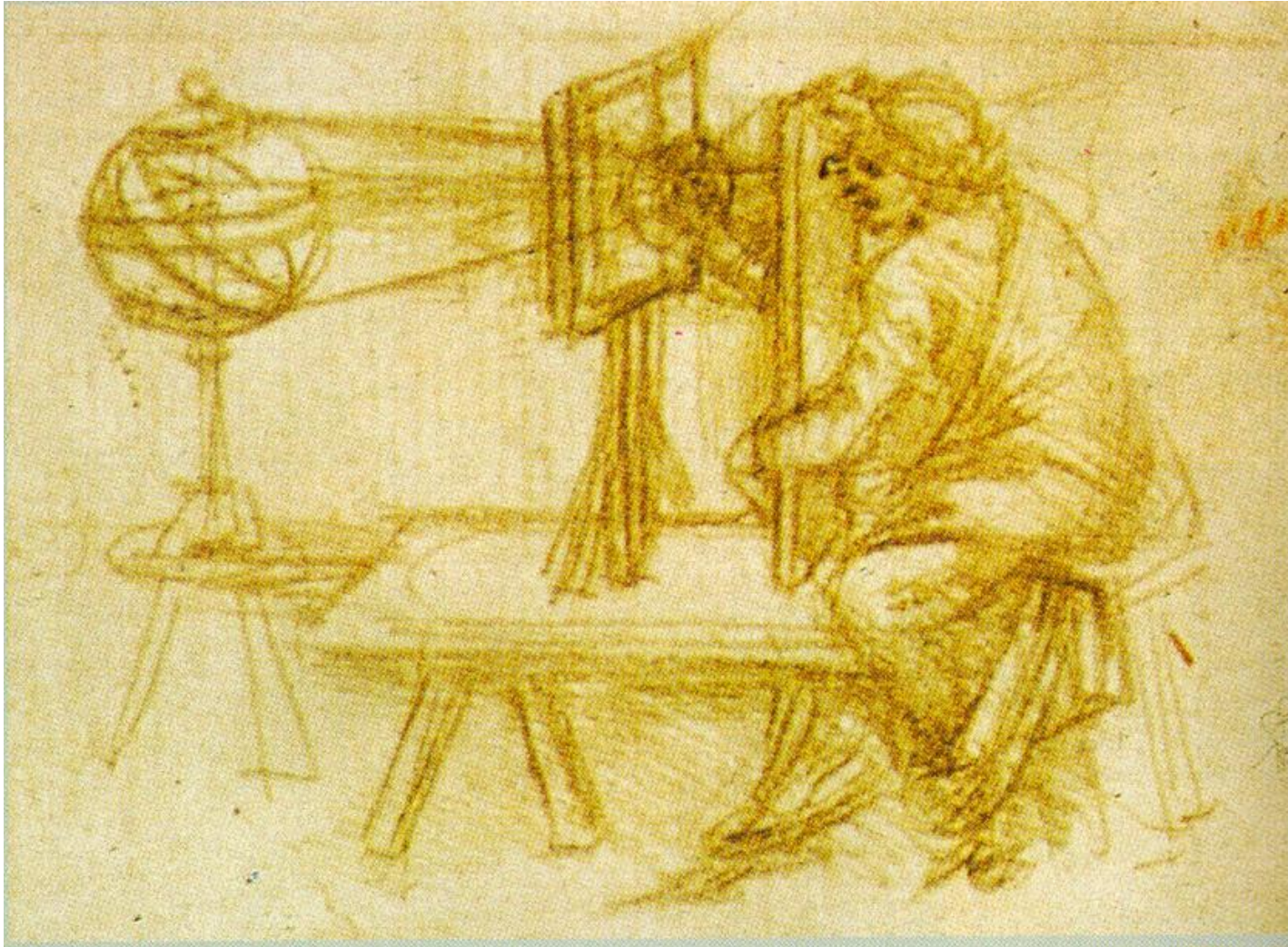
---





# 1 Geometry - Perspective

---



## 2 Probabilistic framework

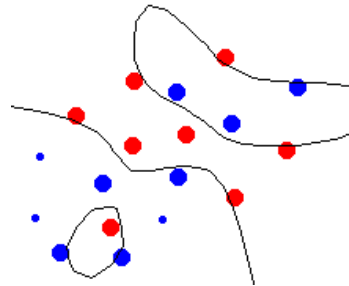
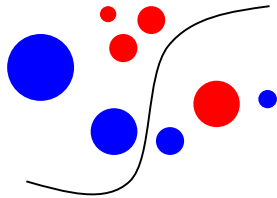
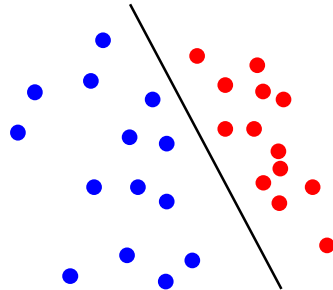
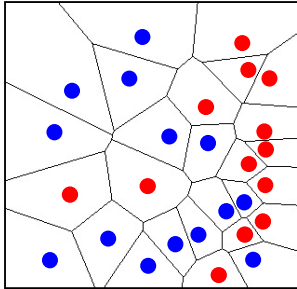
---

“Perception is our best guess as to what is in the world, given our current sensory input and our prior experience.”      Helmholtz (1988)

1. Deal with the ambiguity of the visual world
2. Are able to fuse information
3. Have the ability to learn

# 3 Machine Learning

---



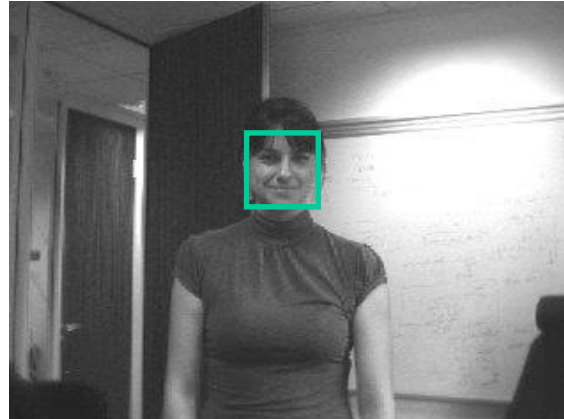
## 2 Computer Vision at Cambridge

# Computer Vision: 3R's

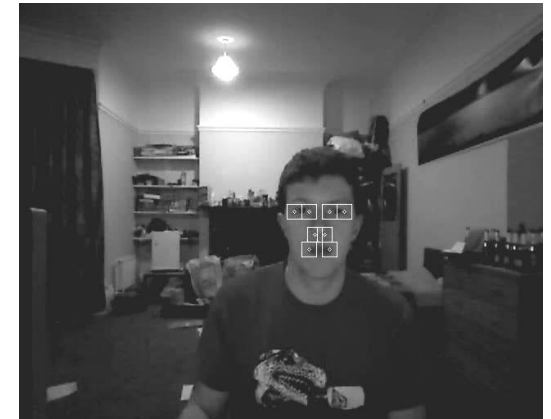
Reconstruction



Recognition



Registration



**Reconstruction:** Recover 3D shape

**Recognition:** Identify objects ([example](#))

**Registration:** Compute their position and pose

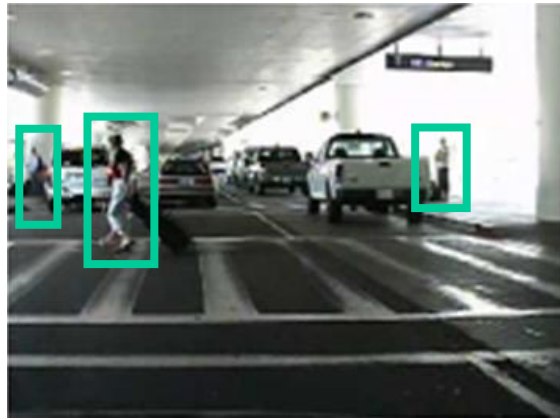


# Computer Vision: 3R's

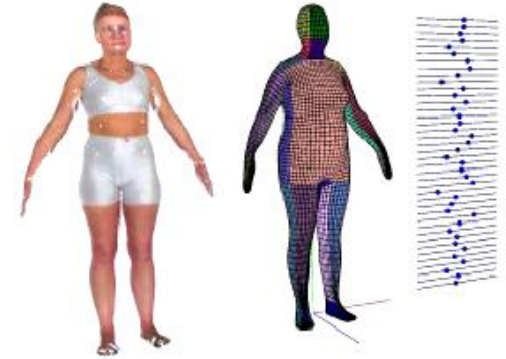
## Reconstruction



## Recognition



## Registration



**Reconstruction:** Recover 3D shape

**Recognition:** Identify objects ([example](#))

**Registration:** Compute their position and pose

# Reconstruction?

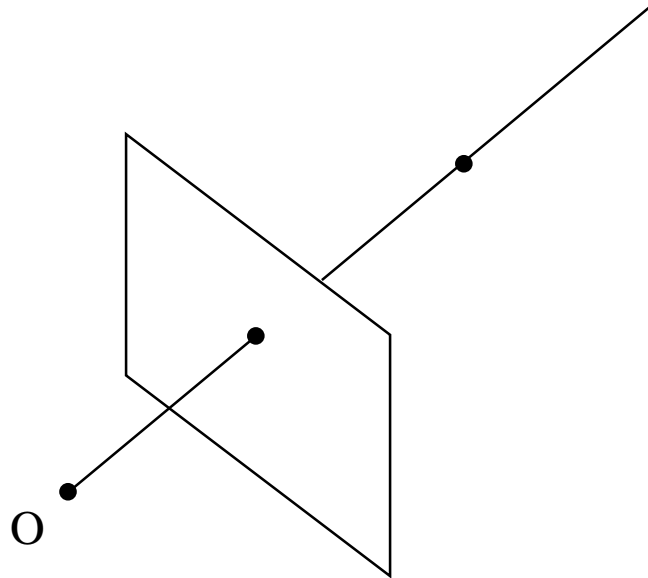
Recovery of 3D shape from  
images

# Reconstruction



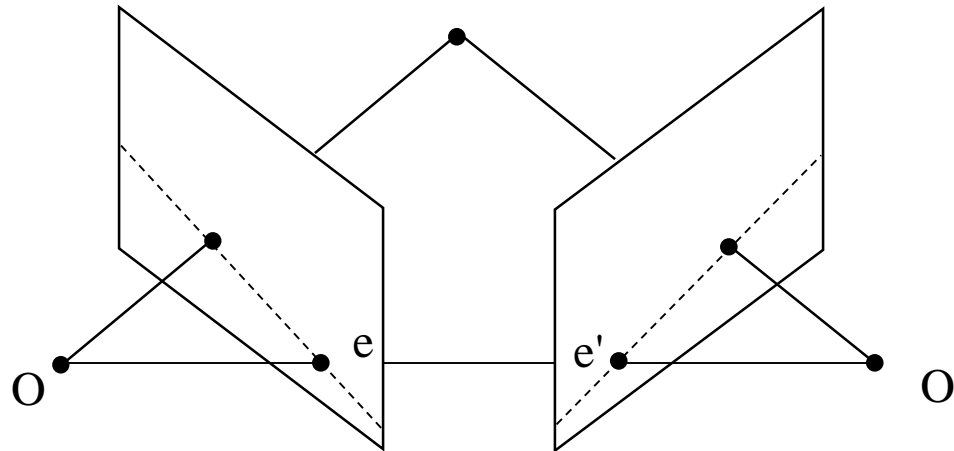
# Ambiguity in a single view

---



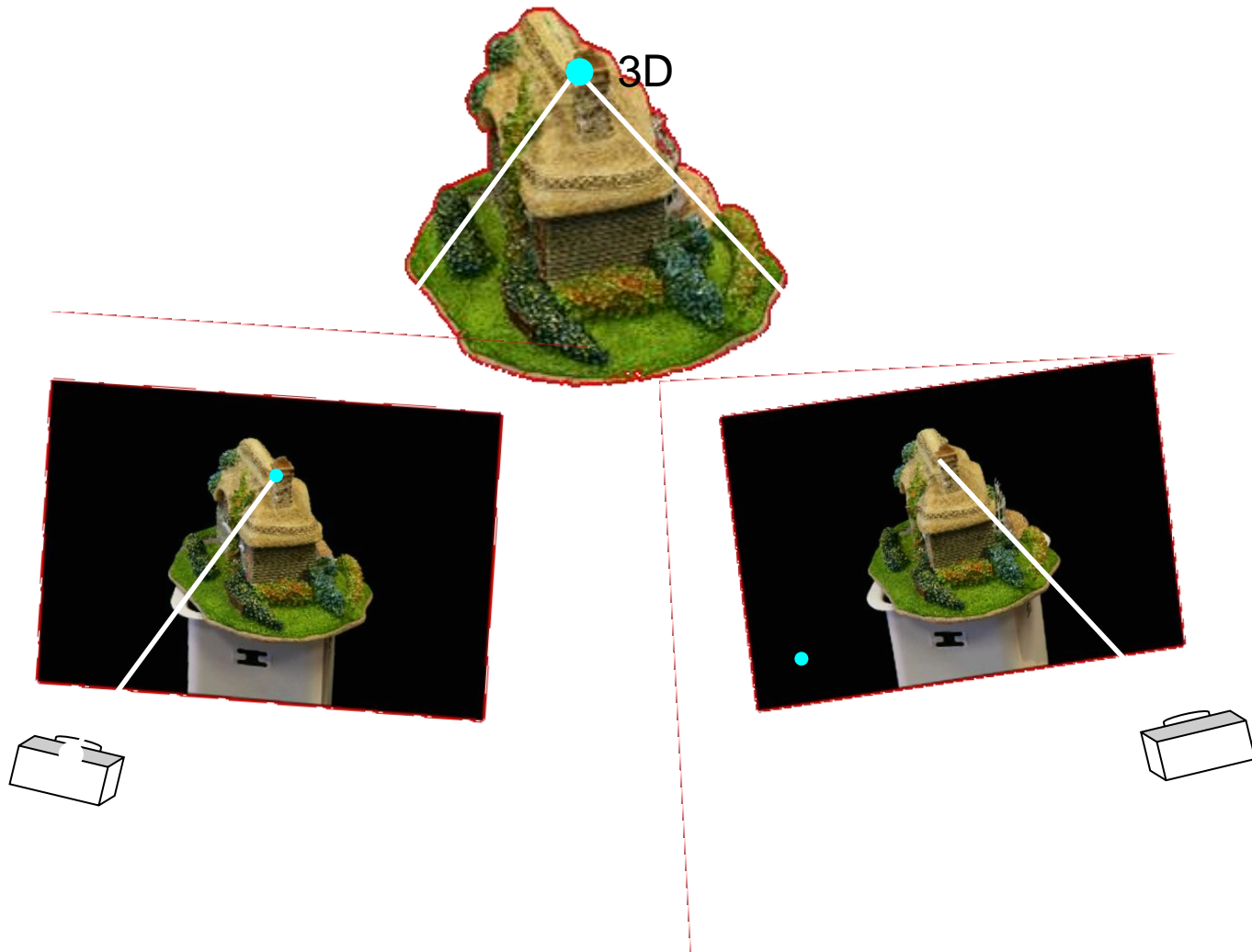
# Stereo vision

---

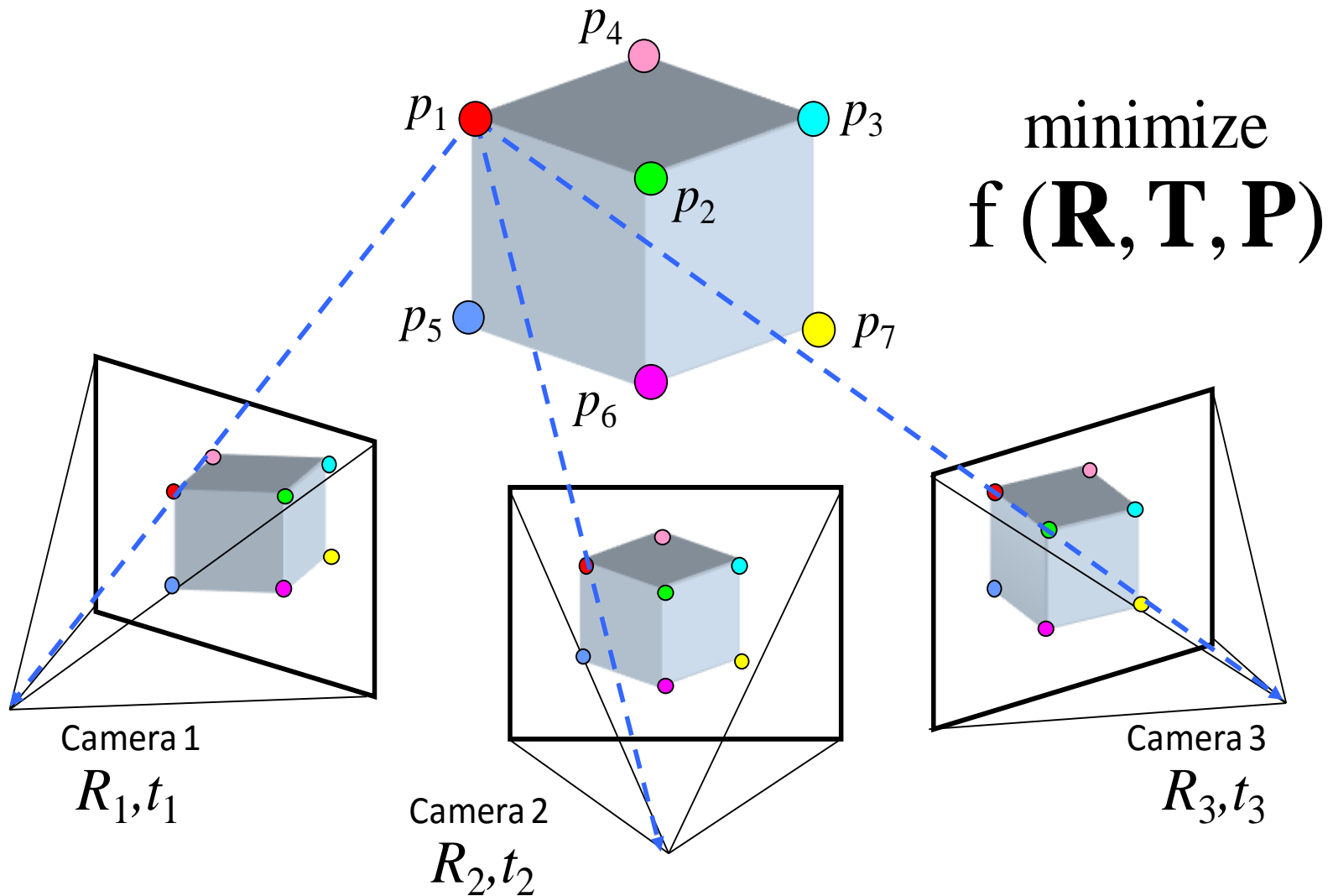




# Stereo vision

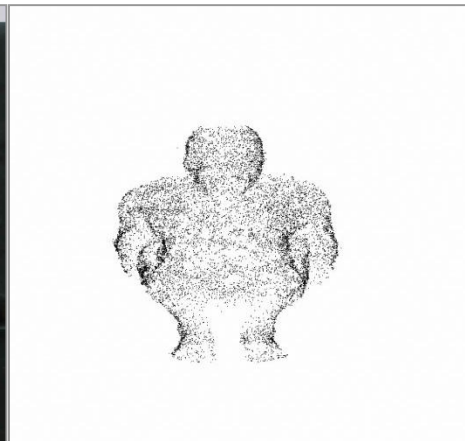
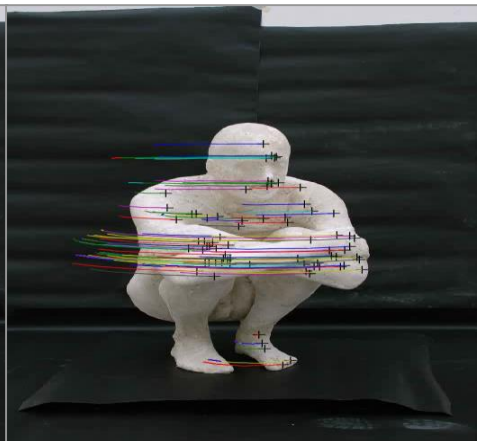
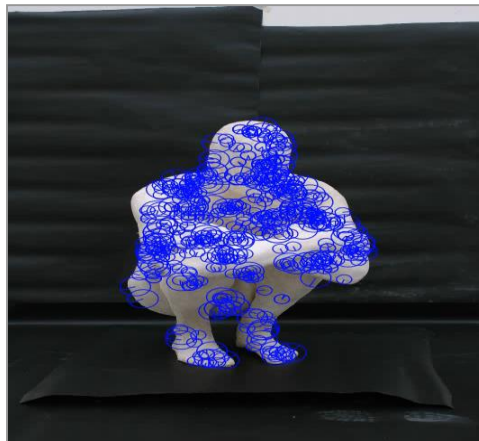


# Multi-view stereo



# Structure from motion

---



Input sequence

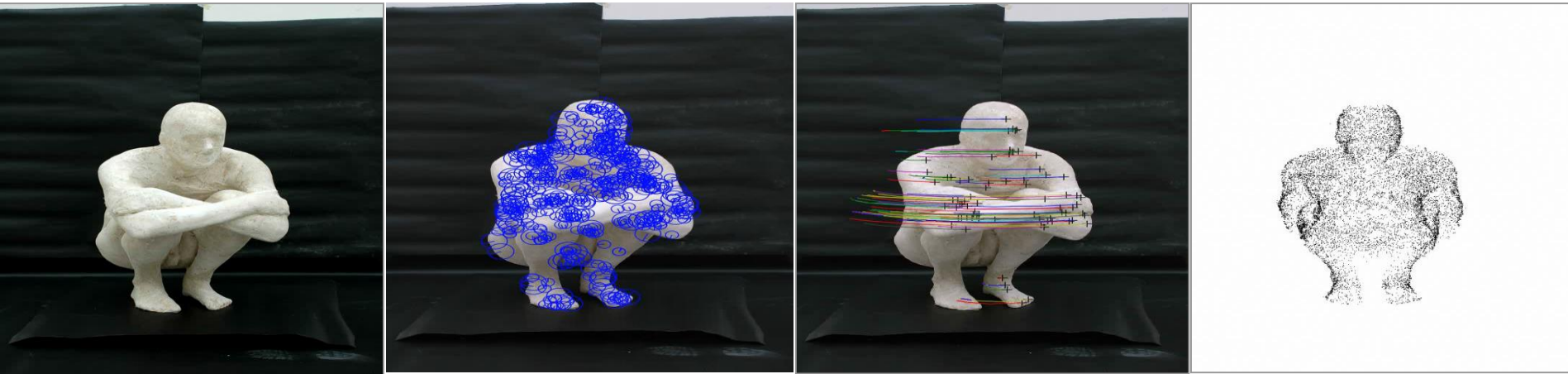
2D features

2D track

3D points

# Structure from motion

---



Input sequence

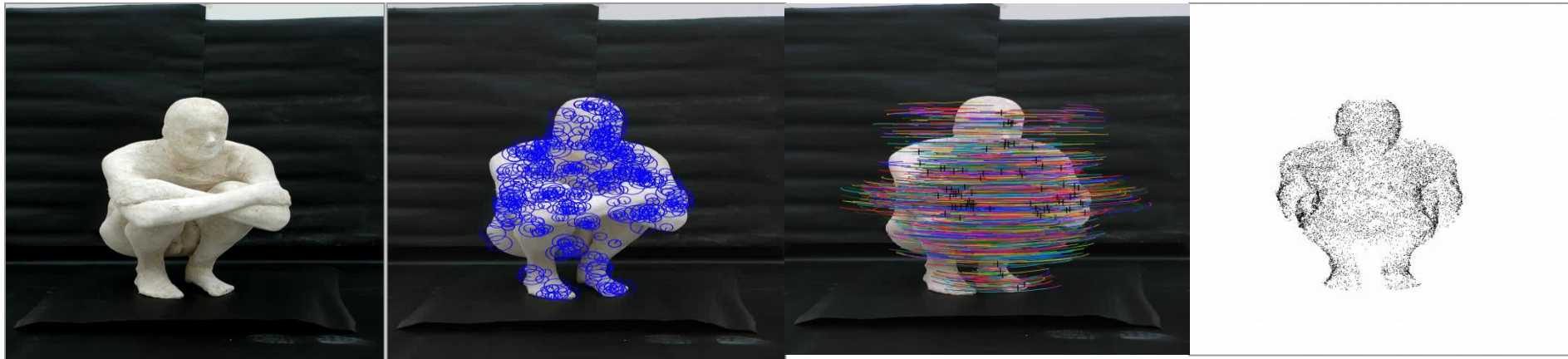
2D features

2D track

3D points

# Structure from motion

---



Input sequence

2D features

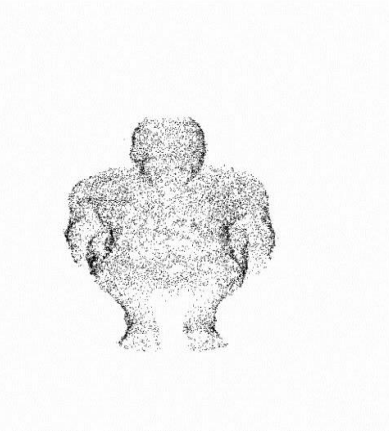
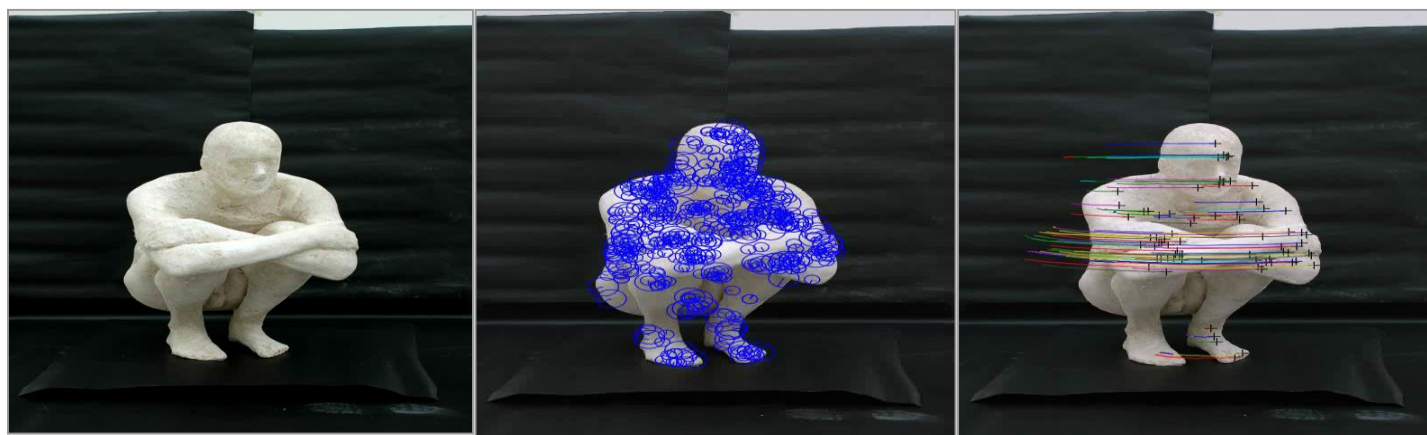
2D track

3D points



# Structure from motion

---



Input sequence

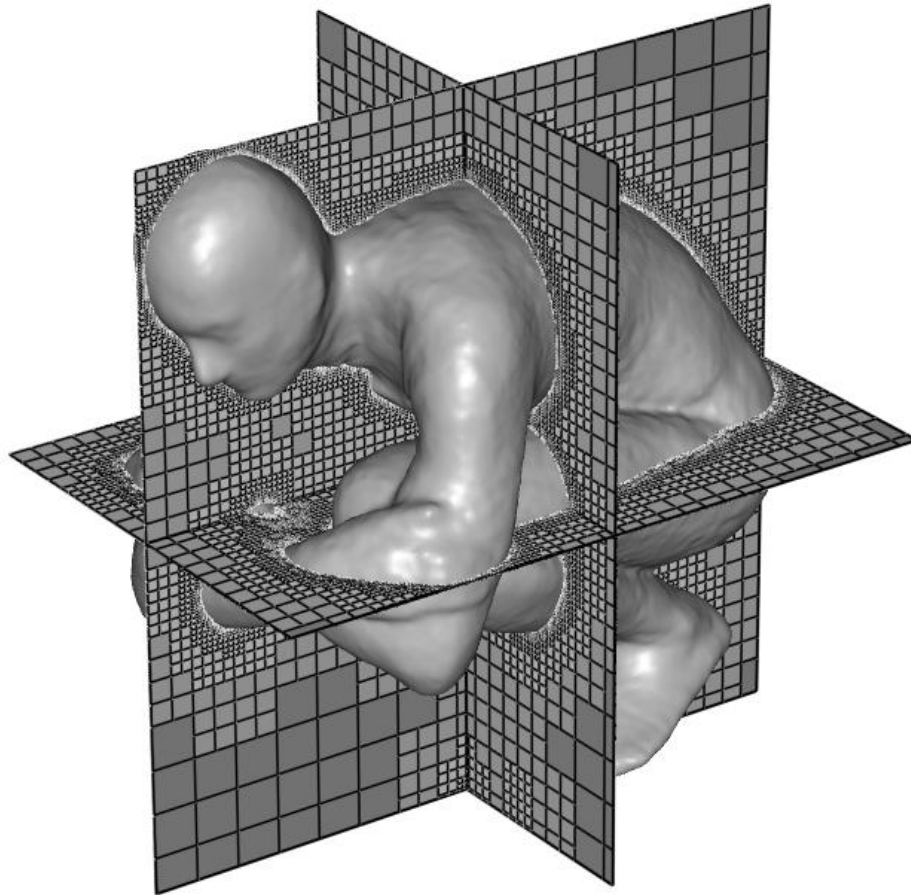
2D features

2D track

3D points

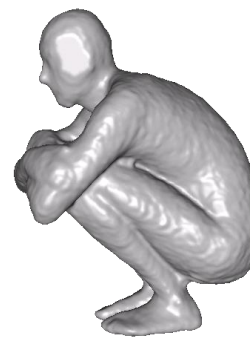
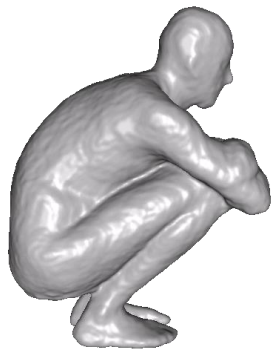
# 3D MRF for 3D modelling

---

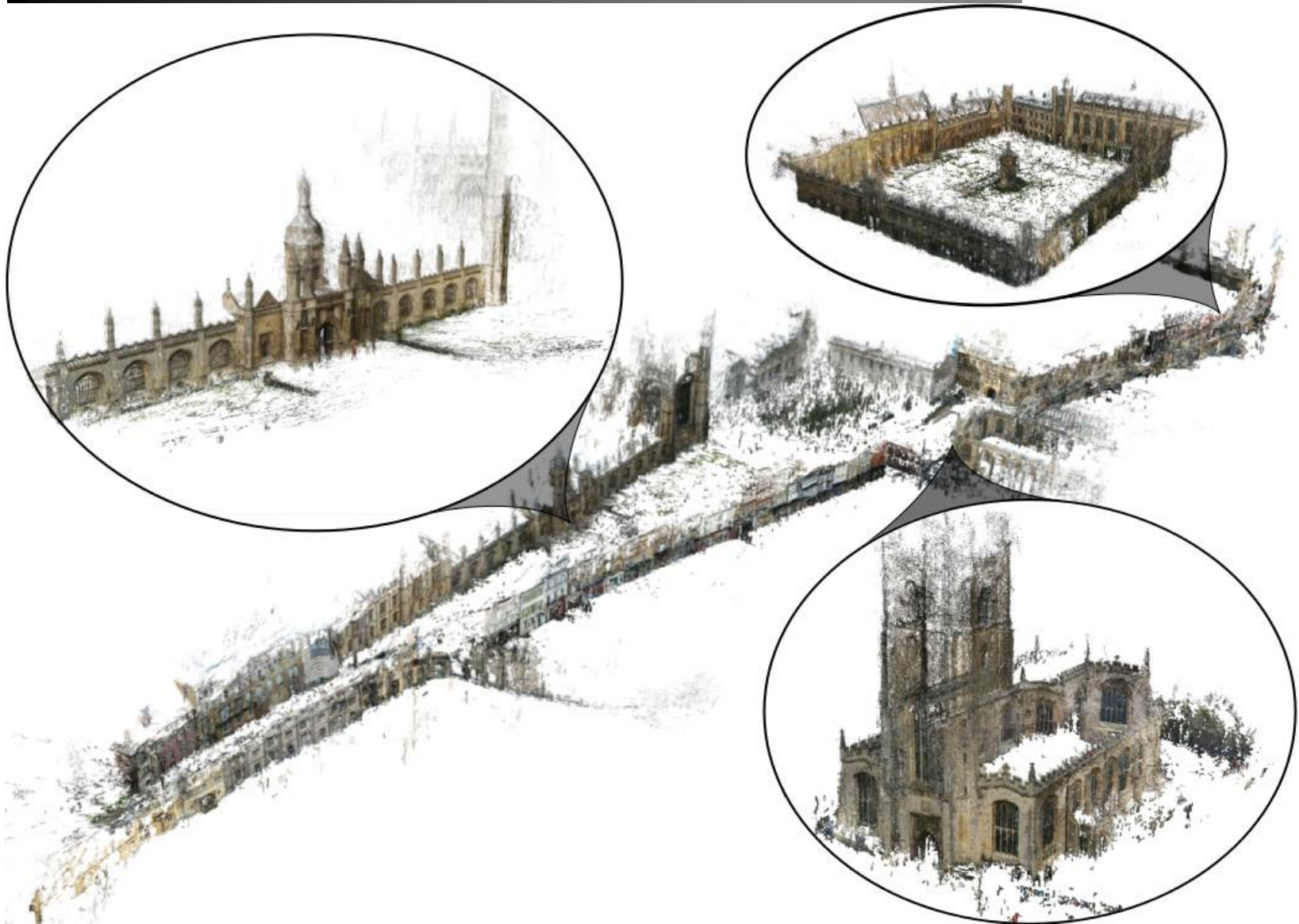


# 3D Models

---



# Large Scale Reconstruction



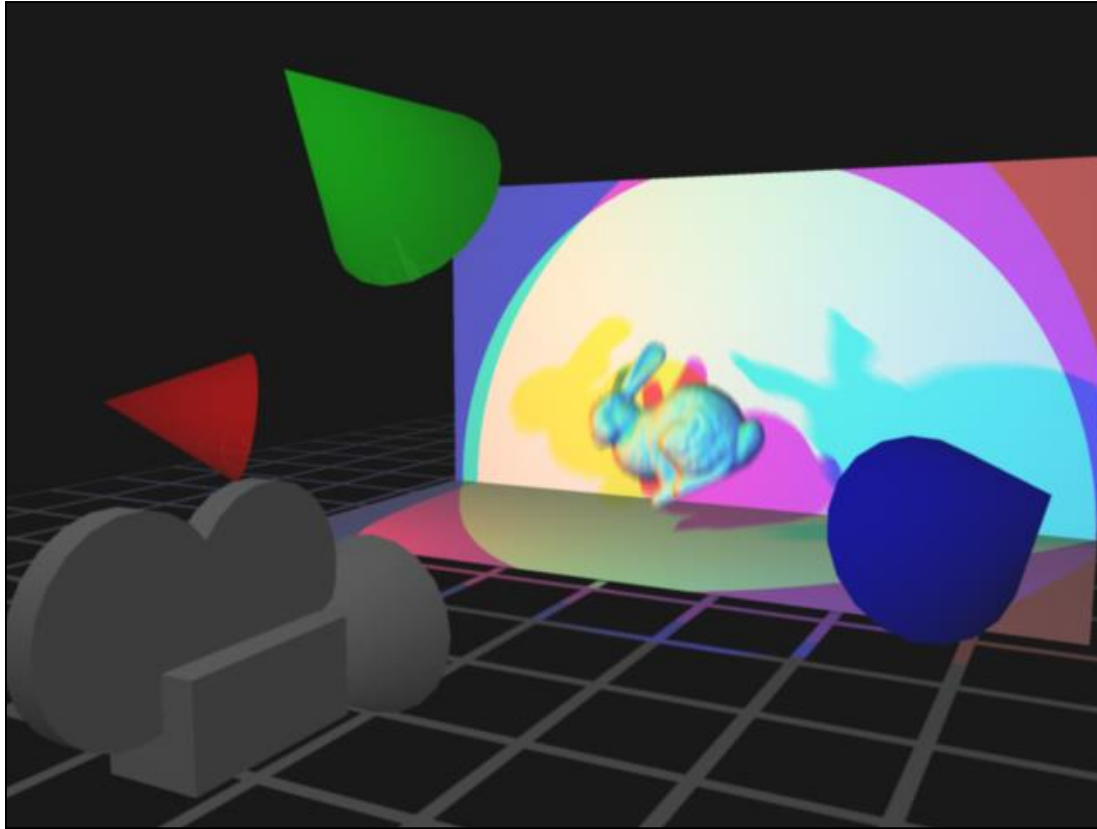
Deformable objects:

Real-time photometric stereo  
using colour lighting



# Textureless deforming objects

---



- a method for reconstructing a textureless *deforming* object in 2.5d

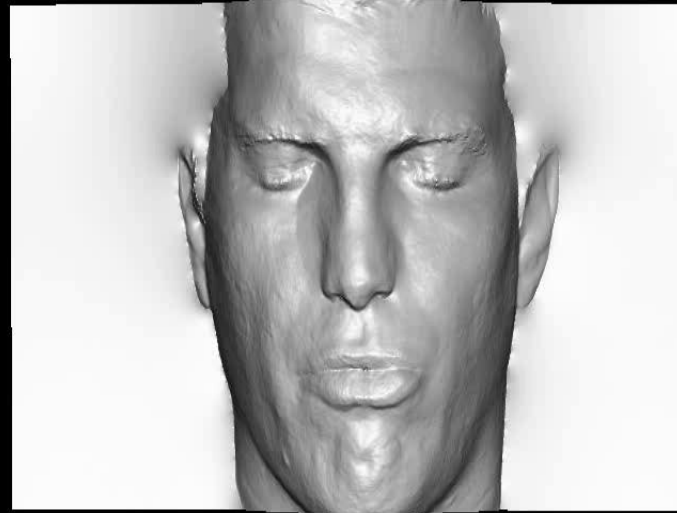
# Colour Photometric Stereo

---



# Real-time deformable surfaces

---



RECORDING  
frame\_rate: 5145.217578

# Sample Reconstructions

## Face capture - Example 1



Original viewpoint



Novel viewpoint

# Registration?

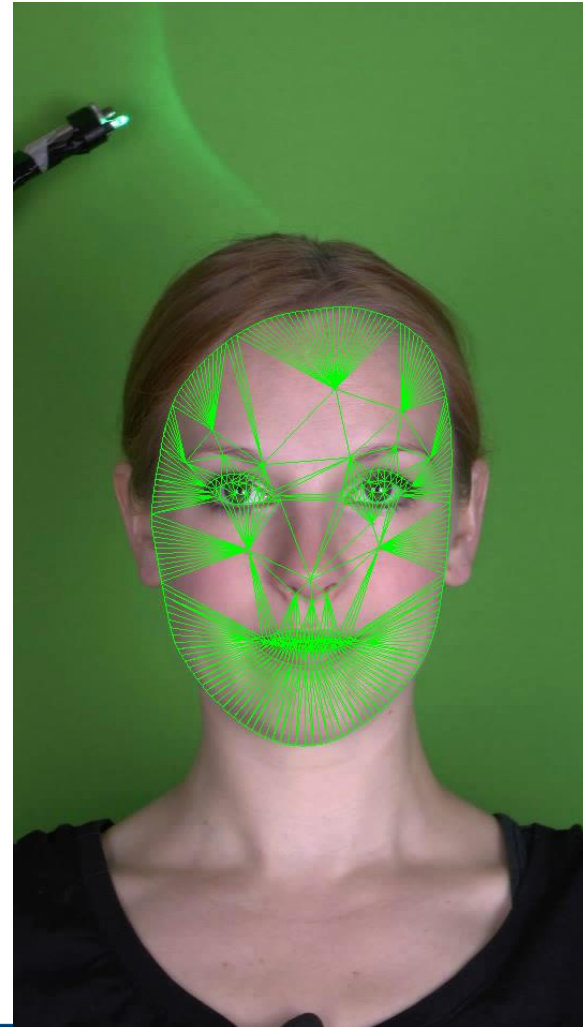
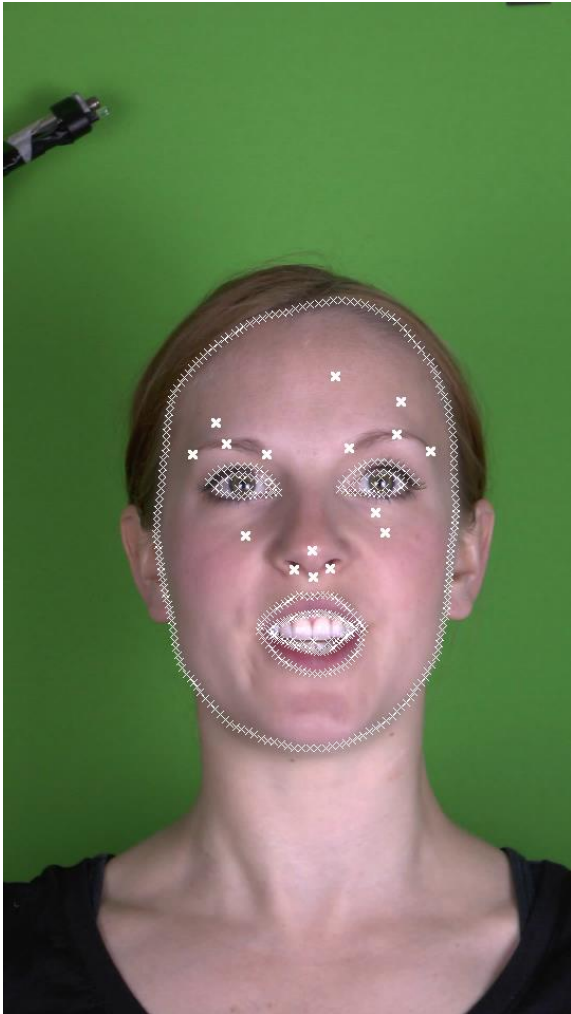
## Target detection and pose estimation



# Registration:

# Expressive Visual Text-to- Speech

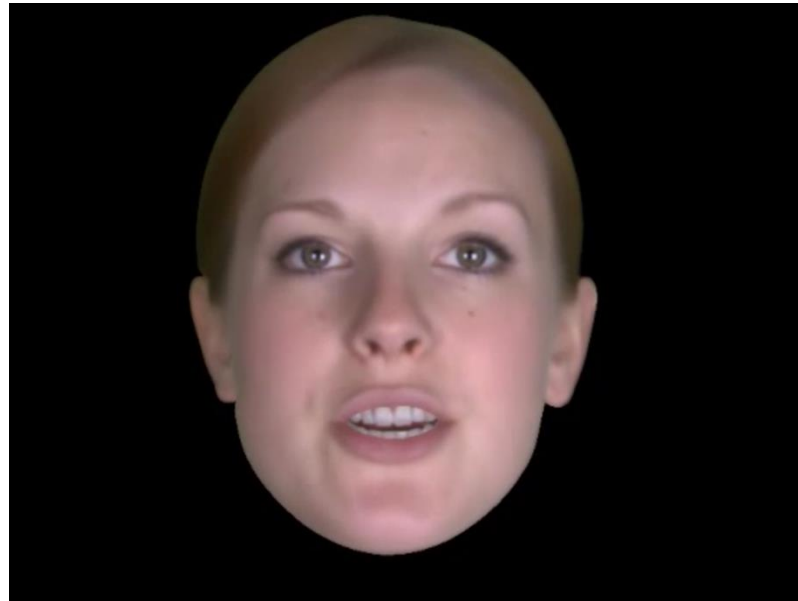
# Registration – alignment of training data



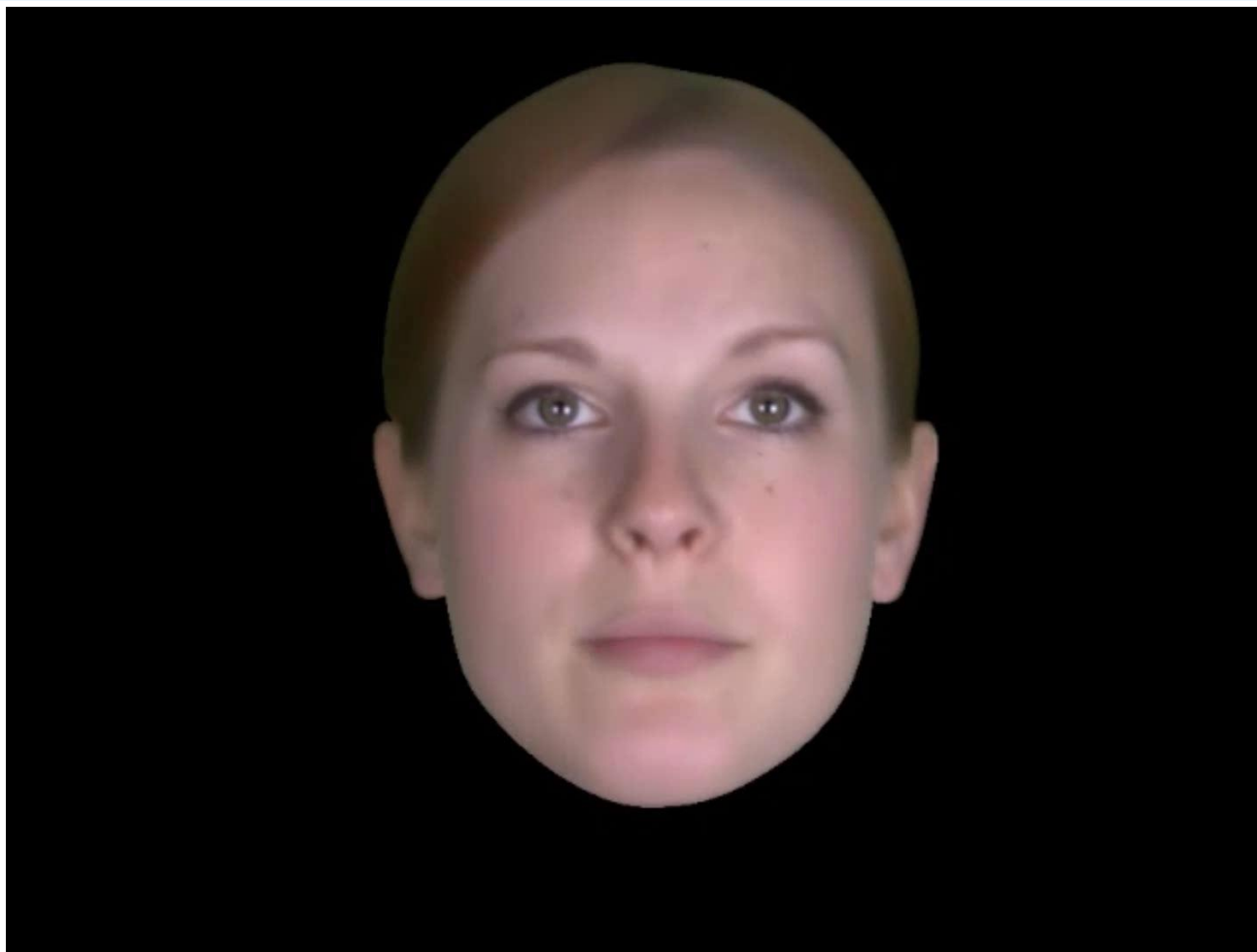
# What is an expressive talking head?

- > User inputs a sentence which they wish to be uttered
- > User specifies an emotion

Video output is generated



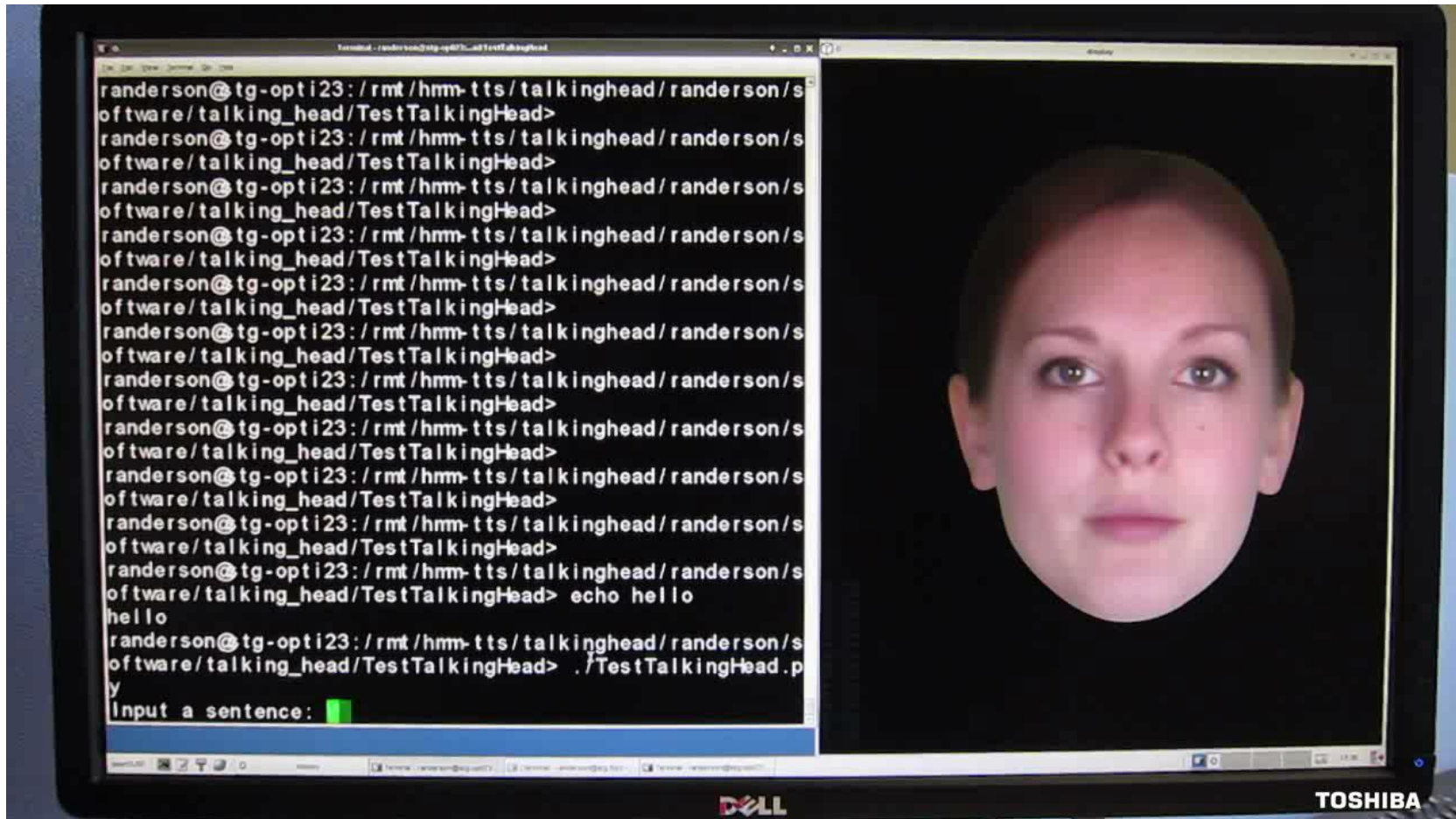
# Our current talking head



# Expressive Visual Text to Speech



# Demo – XpressiveTalk





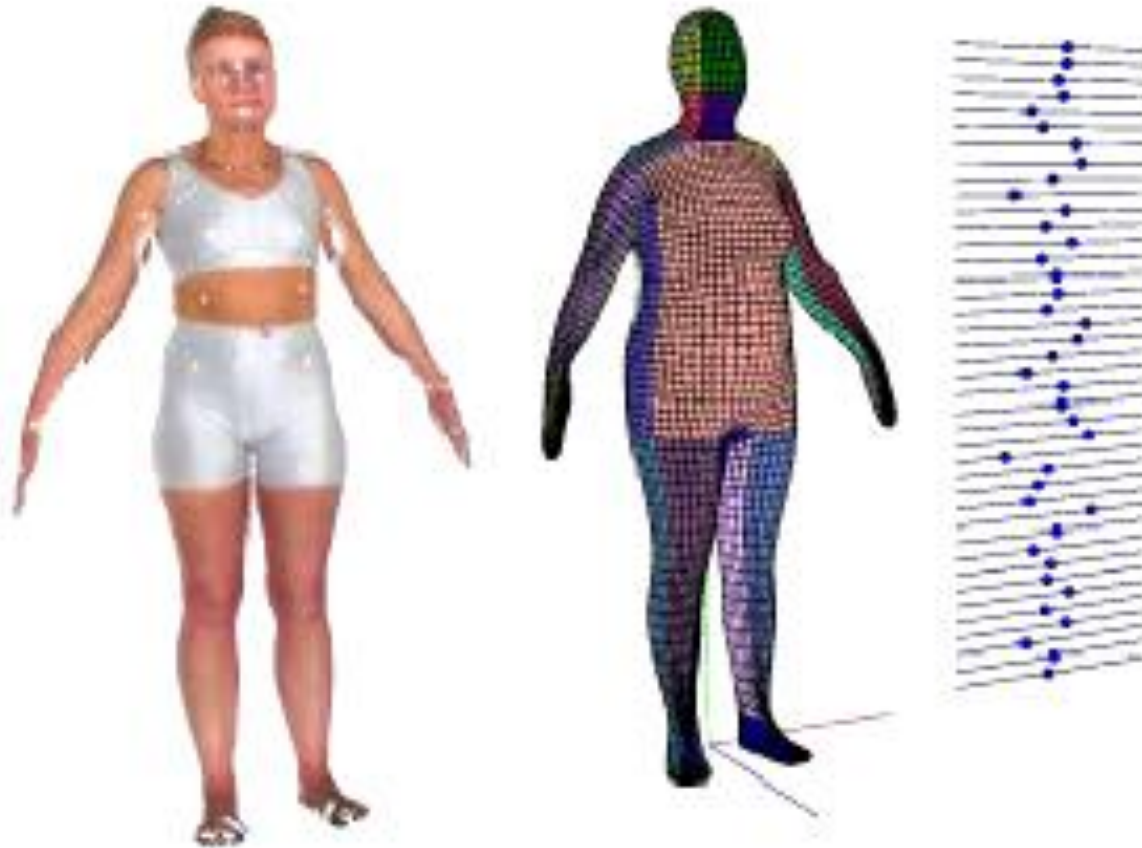
# 3D Registration - Magic Mirrors

---



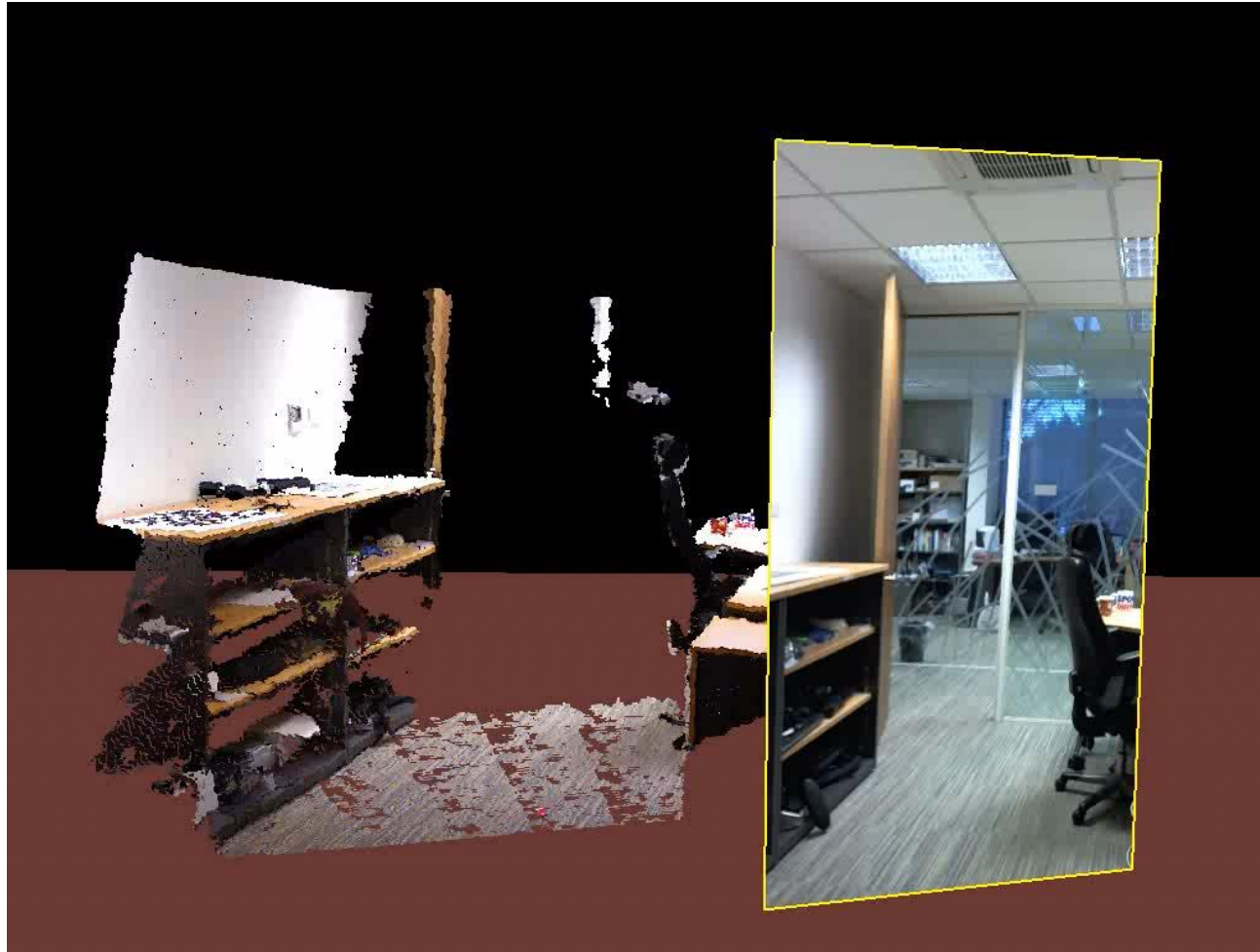
# Registration – Body shape

---



# Single-shot Body Shape

---



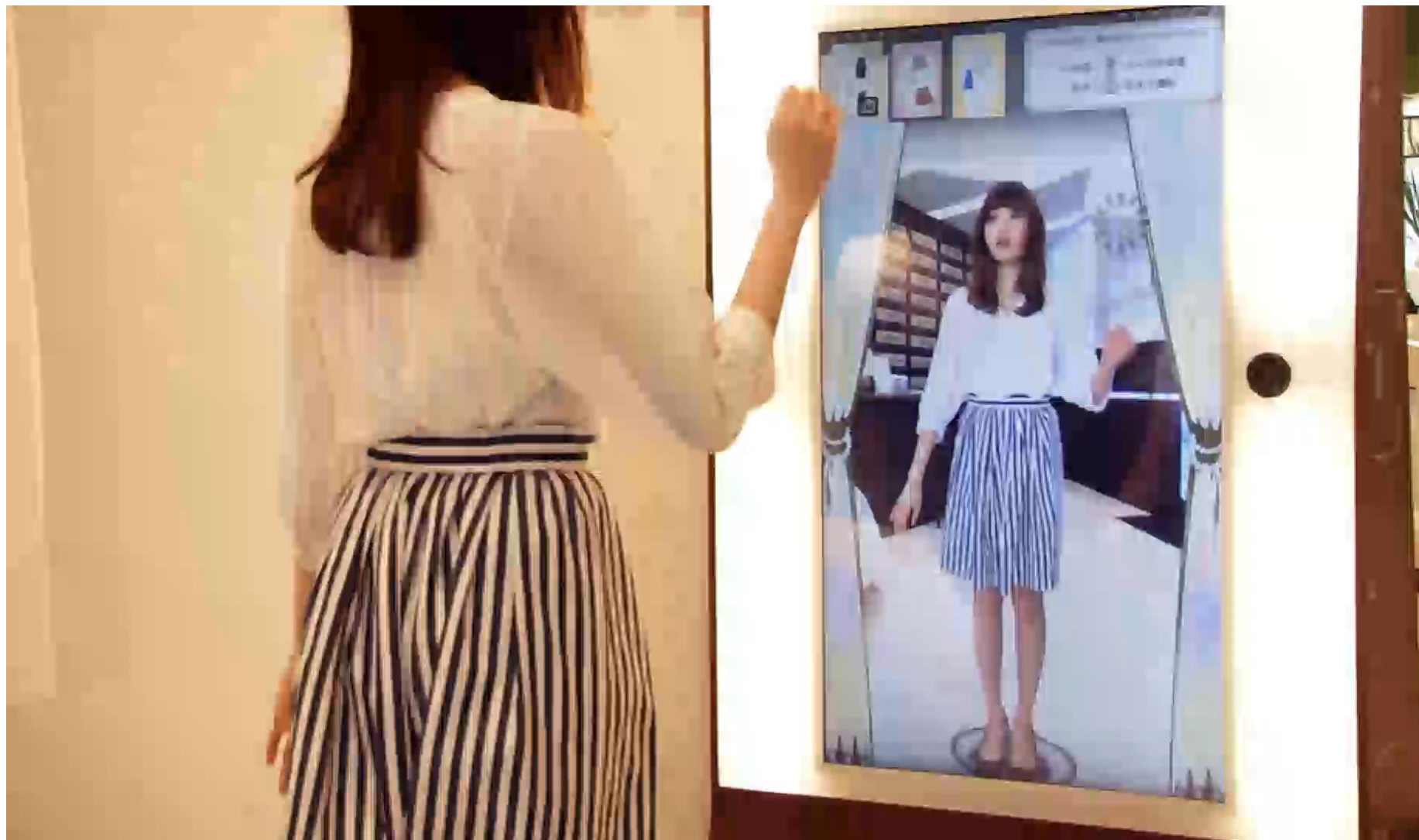
# Single-shot Body Shape

---



# Single-shot Body Shape

---

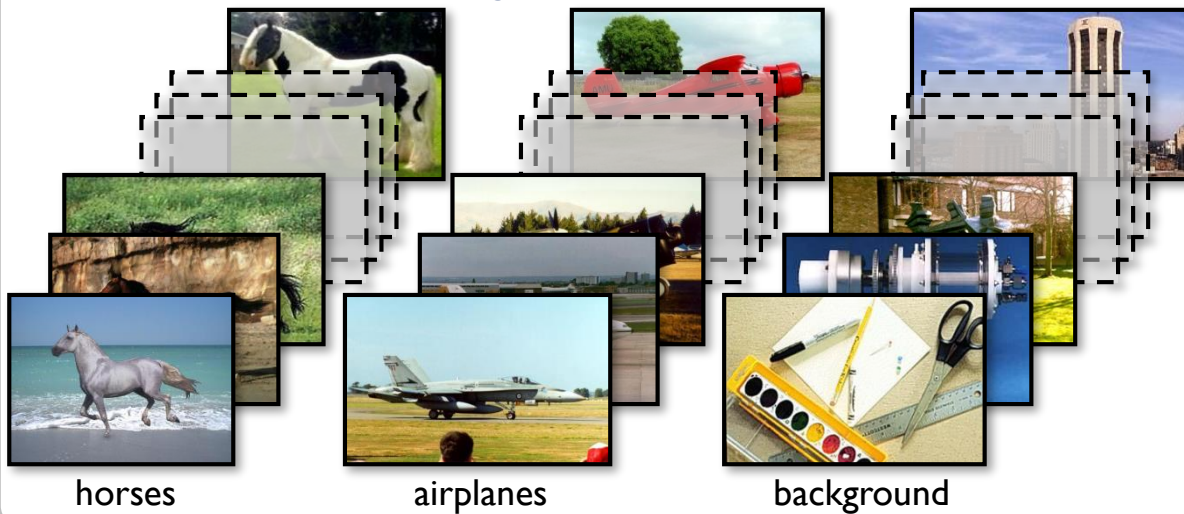


# Recognition?



# Recognition

image classification



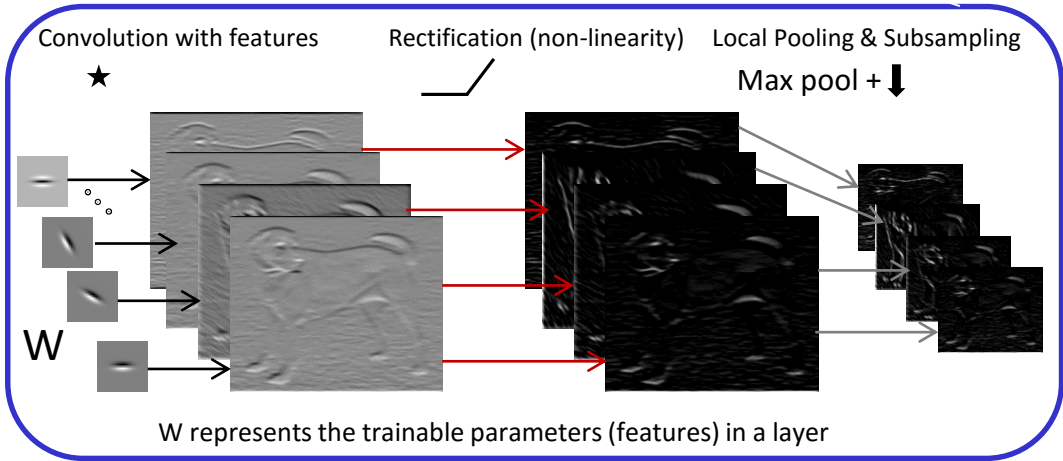
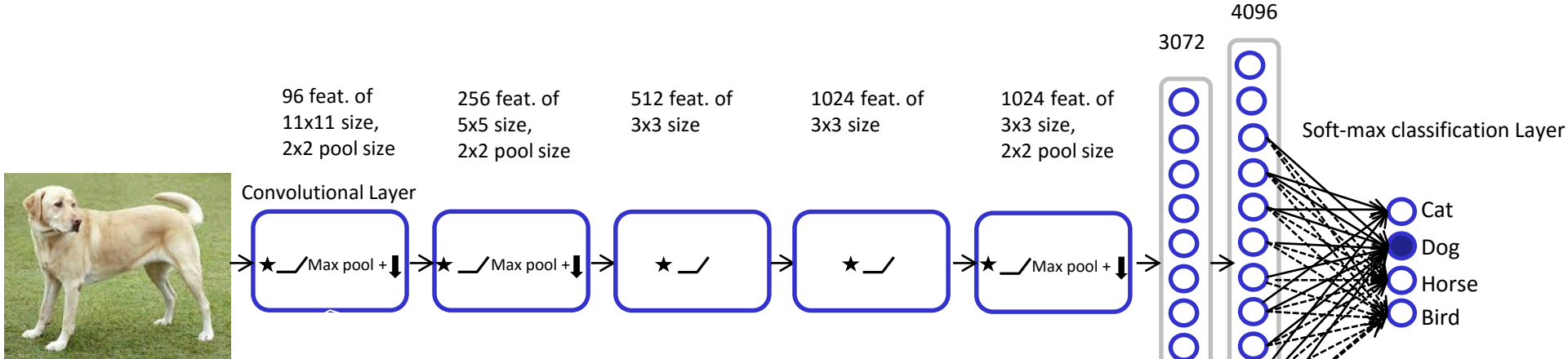
categorical object detection



semantic segmentation



# Deep Learning - Class Recognition with CNN

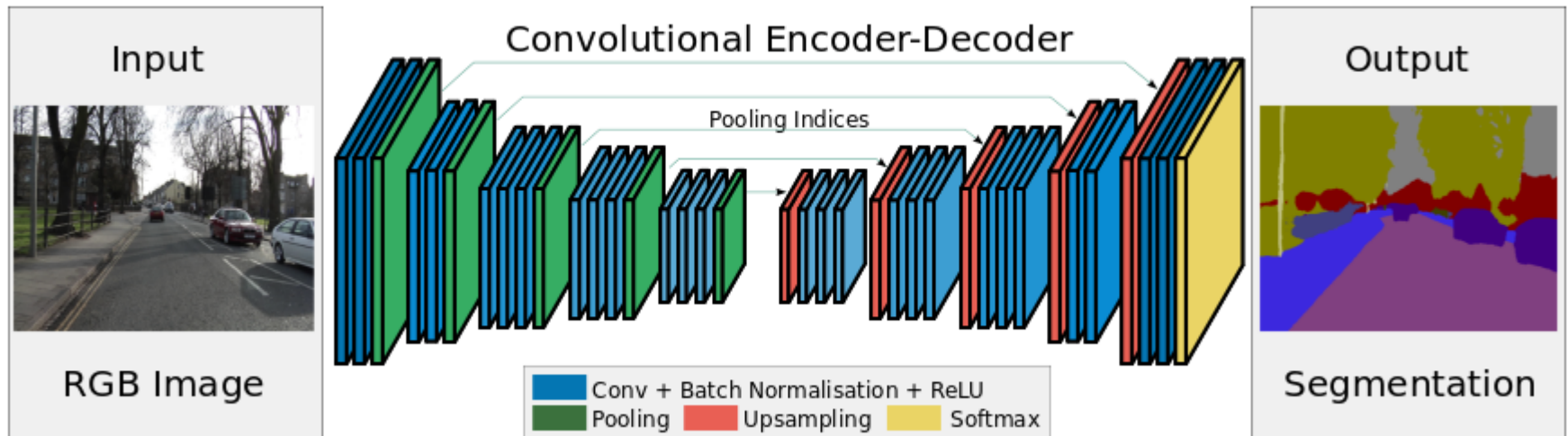


2 fully connected layers

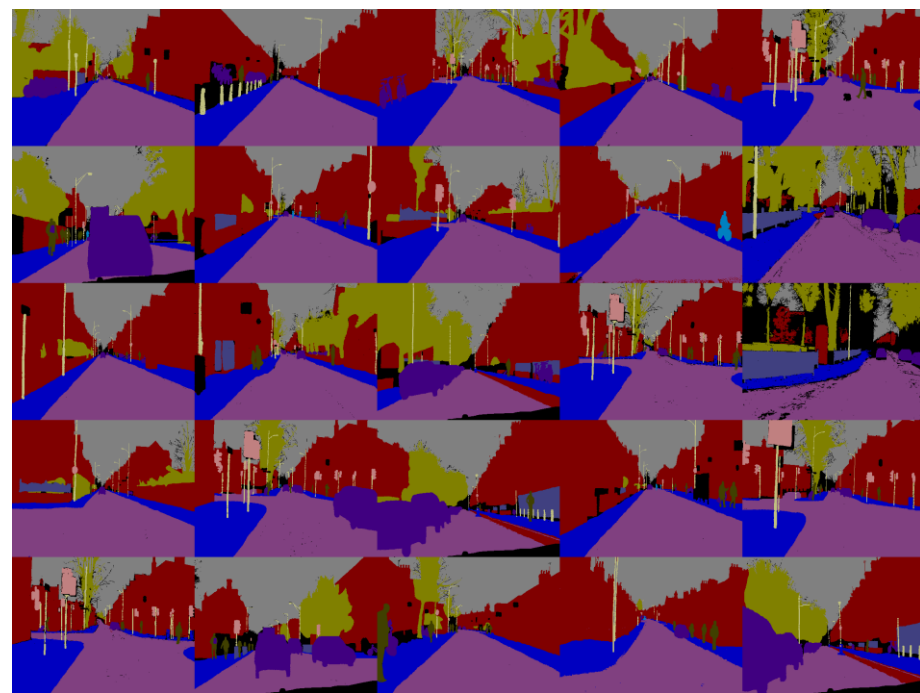
# SegNet Architecture

## Highlights:

- Learns to extract features using an encoder network (e.g. VGG16) and maps features to pixel wise labels using a decoder network.
- Decoders uses the stored pooling indices in the encoding layer to enable upsampling its input to double the resolution.
- Non-linear upsampling using pooling indices maintains shape of categories, and
- Reduces the number of parameters in the decoder network by a large margin as compared to other recent architectures.

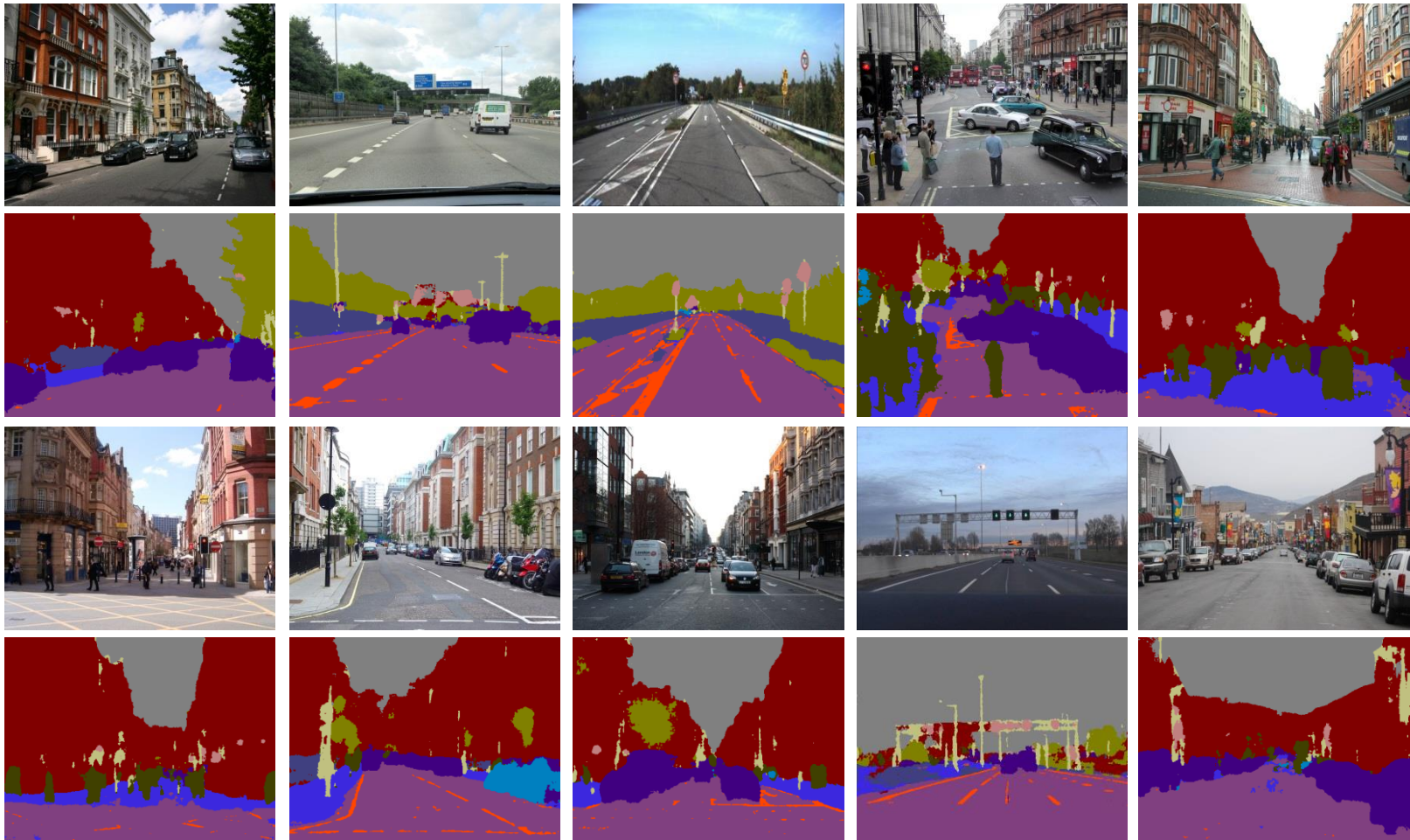


# SegNet – training from labelled data

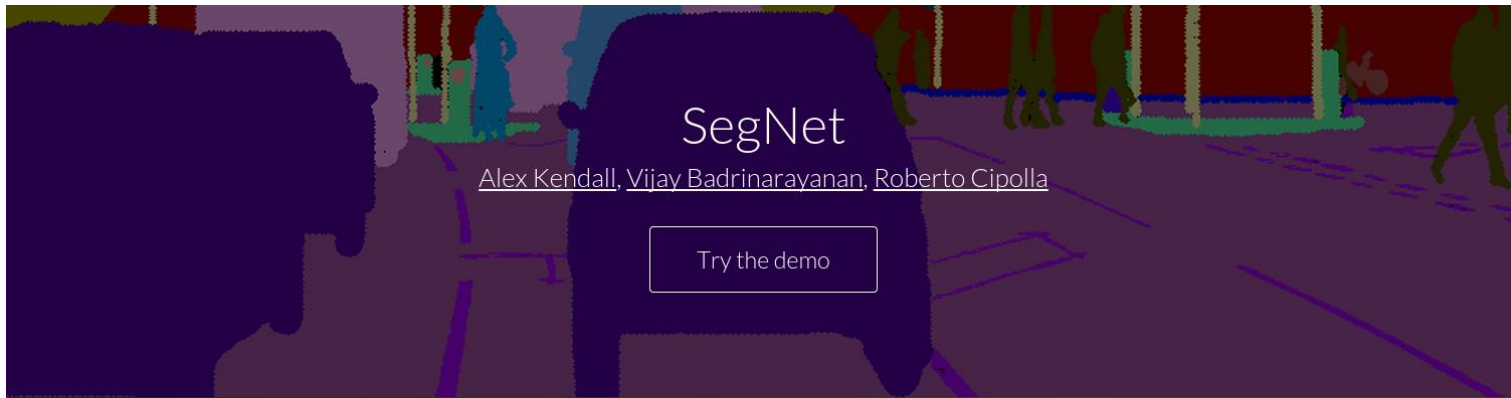




# SegNet predictions on unseen test images - DEMO



# SegNet – Real-time DEMO



SegNet


[Alex Kendall](#), [Vijay Badrinarayanan](#), [Roberto Cipolla](#)

Try the demo


SegNet

A Deep Convolutional Encoder-Decoder Architecture for Robust Semantic Pixel-Wise Labelling

Use a random image, upload your own, search for a place, or click on one of the example images in the gallery below. SegNet is trained to classify each pixel of an urban street image to be one of twelve classes.

Select a Country  

Get Random Image

 Upload an Image File

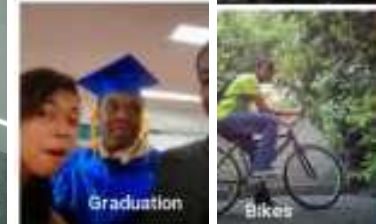
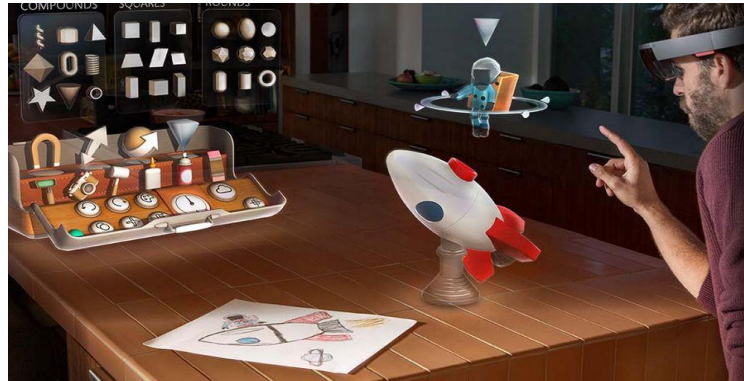
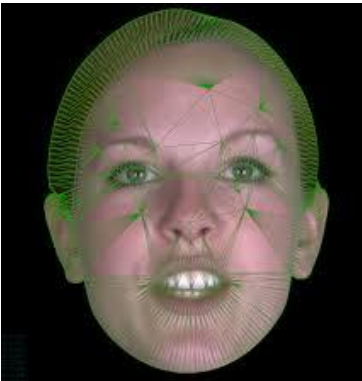
OR Paste Image URL

OR Search for a Place, e.g. Trinity Street Cambridge

Process with SegNet



# Why? Applications



# Summary – Computer Vision

---

1. Background: why and how?
2. 3R's of Computer Vision:
  - Registration
  - Reconstruction
  - Recognition

# More information

---

## Publications:

<http://mi.eng.cam.ac.uk/~cipolla/publications.htm>

## Research demos and code:

<http://mi.eng.cam.ac.uk/projects/segnet/>

<http://mi.eng.cam.ac.uk/projects/relocalisation/>

## Research Videos:

<https://www.youtube.com/user/ComputerVisionVideos>