Joint Attention

through Gaze Following and Motor Babbling

Alessandro Ardu Bram v. Asseldonk Saskia v.d. Wegen

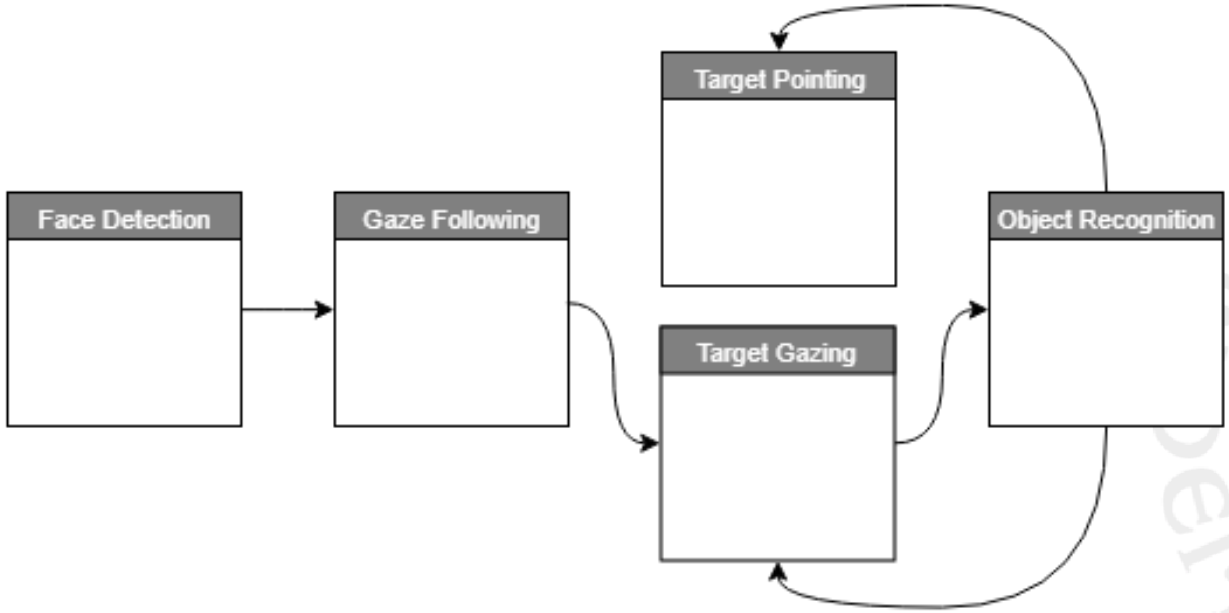
s4793242 s4476883 s4495020

SOW-BKI57 Developmental Robotics
Radboud University Nijmegen

Demo Day Presentation
27th June 2018

Outline

Pipeline


Face Detection
Gaze Following
Target Pointing
Target Gazing
Object Recognition

Face detection

- Nao has in-build face detection

Face detection

- Nao has in-build face detection
- openCV haarcascade frontal faces classifier

Face detection

- Nao has in-build face detection
- openCV haarcascade frontal faces classifier
- openCV haarcascade eyes classifier

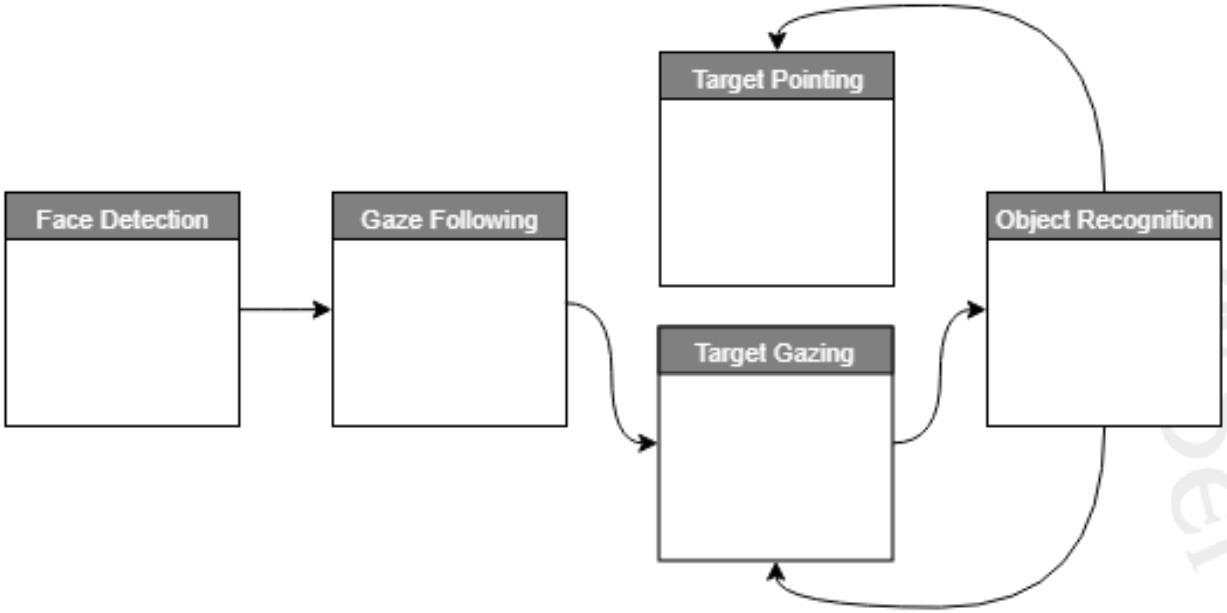
Face detection

- Nao has in-build face detection
- openCV haarcascade frontal faces classifier
- openCV haarcascade eyes classifier
- openCV haarcascade profile faces classifier

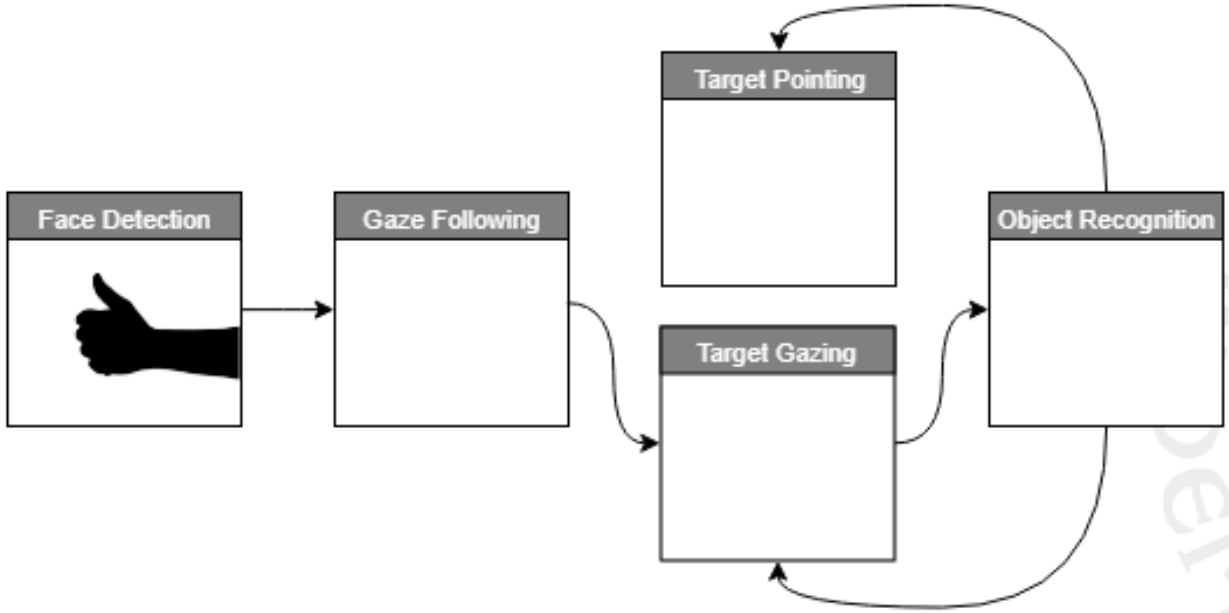
Face detection

- Nao has in-build face detection, which we didn't use
- openCV haarcascade frontal faces classifier
- openCV haarcascade eyes classifier
- openCV haarcascade profile faces classifier

Pipeline


Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Pipeline


Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Gaze following

- Pieters code for the gaze-following model[1][2]

Gaze following

- Pieters code for the gaze-following model[1][2], but Caffe

Gaze following

- Pieters code for the gaze-following model[1][2], but Caffe
- Ported the model to chainer

Gaze following

- Pieters code for the gaze-following model[1][2], but Caffe
- Ported the model to chainer
- Many tears, hours of effort and magic

Gaze following

- Pieters code for the gaze-following model[1][2], but Caffe
- Ported the model to chainer
- Many tears, hours of effort and magic
- We get predictions!

Gaze following

- Pieters code for the gaze-following model[1][2], but Caffe
- Ported the model to chainer
- Many tears, hours of effort and magic
- We get predictions! **YAY!**

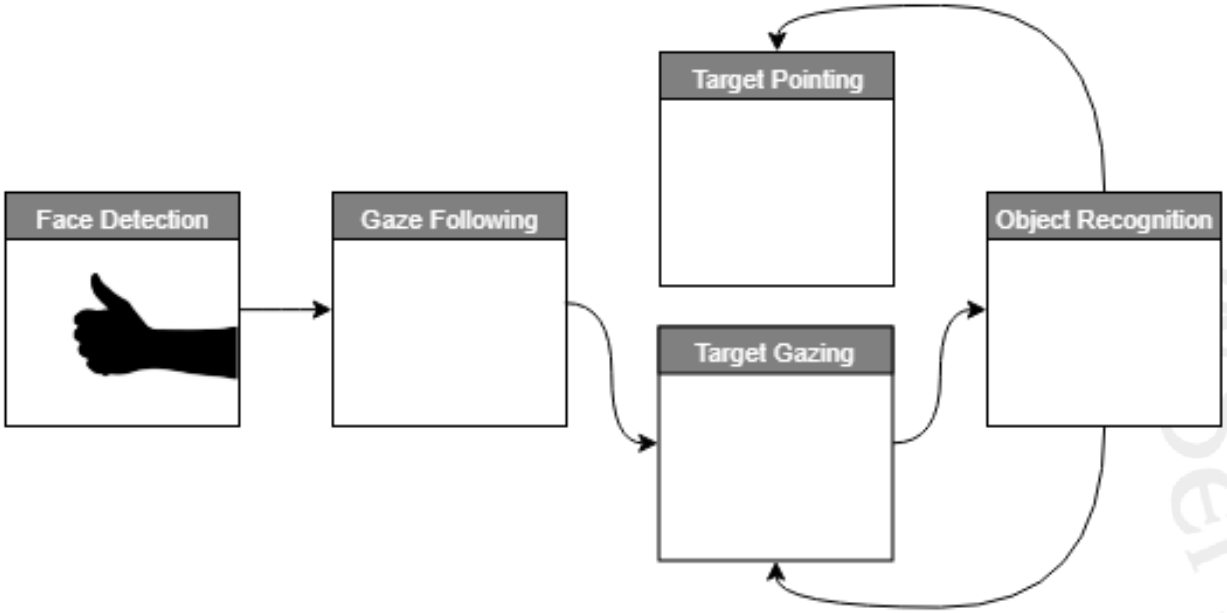
Gaze following

- Pieters code for the gaze-following model[1][2], but Caffe
- Ported the model to chainer
- Many tears, hours of effort and magic
- We get predictions! Just not very accurate ones...

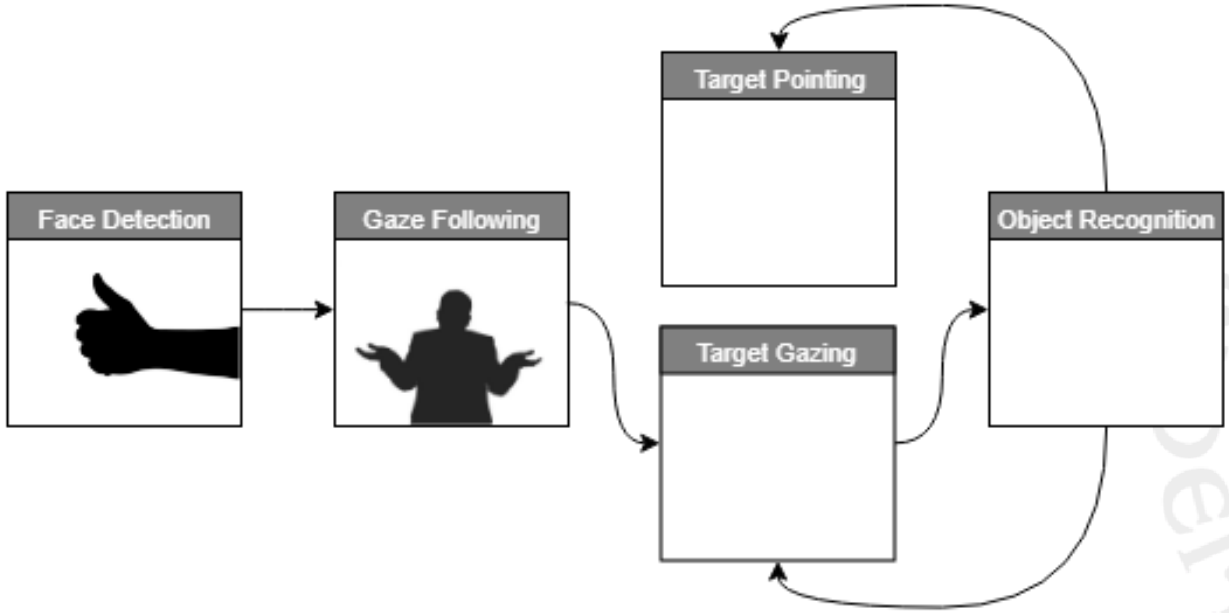
Gaze following

- Pieters code for the gaze-following model[1][2], but Caffe
- Ported the model to chainer
- Many tears, hours of effort and magic
- We get predictions! Just not very accurate ones... :/

Pipeline


Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Pipeline


Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Target gazing

- **Input:** pixel coordinates of predicted gaze location

Target gazing

- **Input:** pixel coordinates of predicted gaze location
- Pretty basic transformations (e.g. pos to angles)

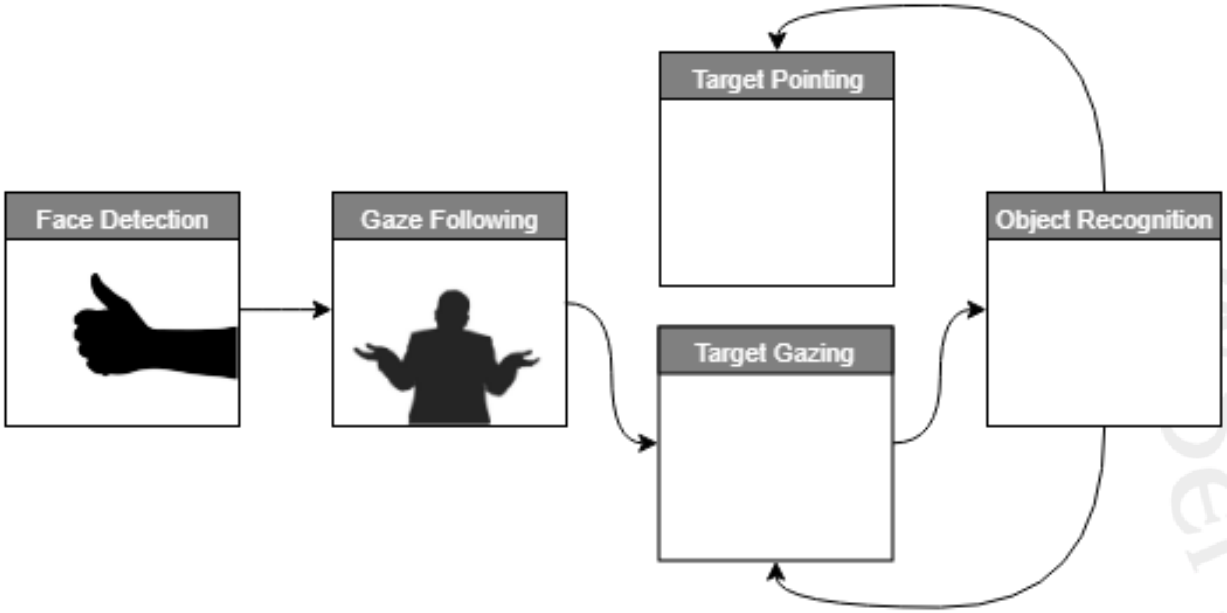
Target gazing

- **Input:** pixel coordinates of predicted gaze location
- Pretty basic transformations (e.g. pos to angles)
- **Output:** radians for the neck joints

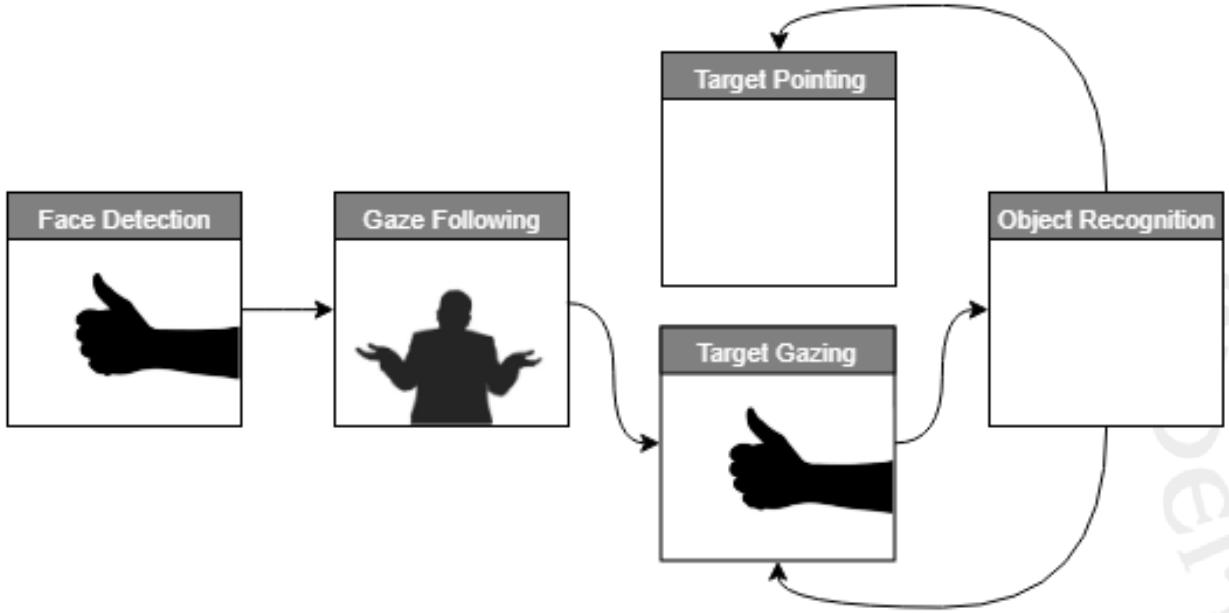
Target gazing

- **Input:** pixel coordinates of predicted gaze location
- Pretty basic transformations (e.g. pos to angles)
- **Output:** radians for the neck joints
- Keeps the limits of the actuators in mind

Pipeline


Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Pipeline


Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Object recognition

- **Input:** image

Object recognition

- **Input:** image
- openCV mask for multiple color ranges

Object recognition

- **Input:** image
- openCV mask for multiple color ranges
- openCV Hough transformation

Object recognition

- **Input:** image
- openCV mask for multiple color ranges
- openCV Hough transformation
- **Output:** centres of detected circles

Object recognition

- **Input:** image

Object recognition

- **Input:** image
- Enhance the contrast

Object recognition

- **Input:** image
- Enhance the contrast
- openCV Median blur

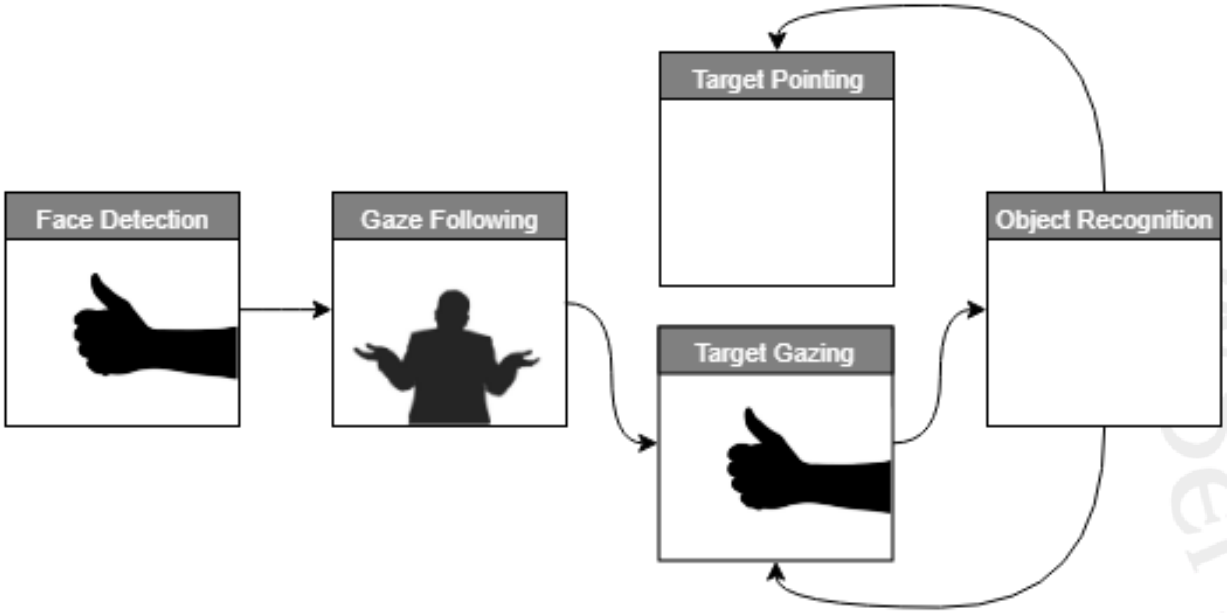
Object recognition

- **Input:** image
- Enhance the contrast
- openCV Median blur
- openCV Hough transformation

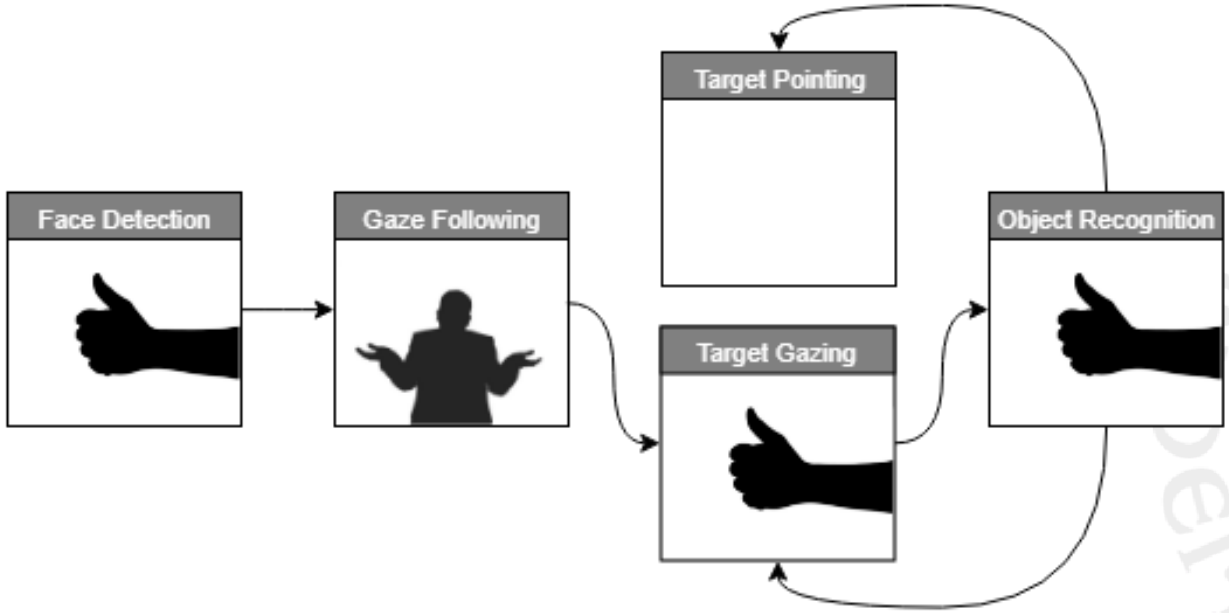
Object recognition

- **Input:** image
- Enhance the contrast
- openCV Median blur
- openCV Hough transformation
- **Output:** centres of detected circles

Pipeline


Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Pipeline


Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Target pointing

- **Input:** pixel coordinates of predicted target location

Target pointing

- **Input:** pixel coordinates of predicted target location
- Pretty basic transformations (e.g. pos to angles)

Target pointing

- **Input:** pixel coordinates of predicted target location
- Pretty basic transformations (e.g. pos to angles)
- **Output:** radians for the arm joints

Target pointing

- **Input:** pixel coordinates of predicted target location
- Pretty basic transformations (e.g. pos to angles)
- **Output:** radians for the arm joints
- Keeps the limits of the actuators in mind

Target pointing

- **Input:** pixel coordinates of predicted target location
- Pretty basic transformations (e.g. pos to angles)
- **Output:** radians for the arm joints
- Keeps the limits of the actuators in mind

Except this is developmental robotics

Target pointing

- Inspiration from Doniec, Sun and Scassellati[3][4]

Except this is developmental robotics

Target pointing

- Inspiration from Doniec, Sun and Scassellati[3][4]
- Motor babbling

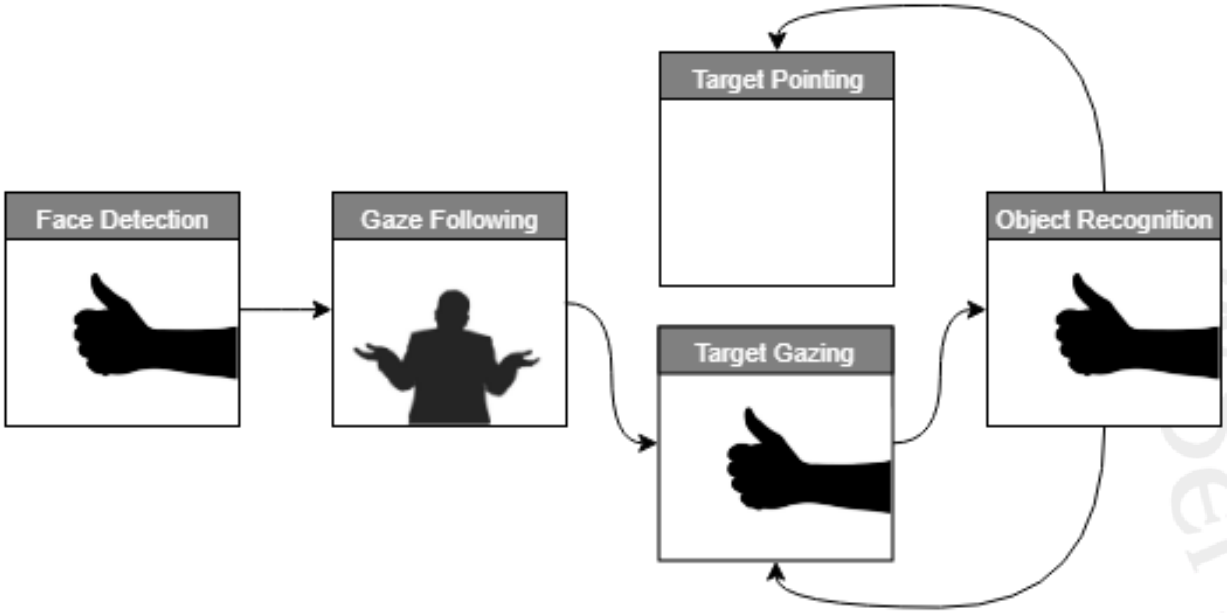
Except this is developmental robotics

Target pointing

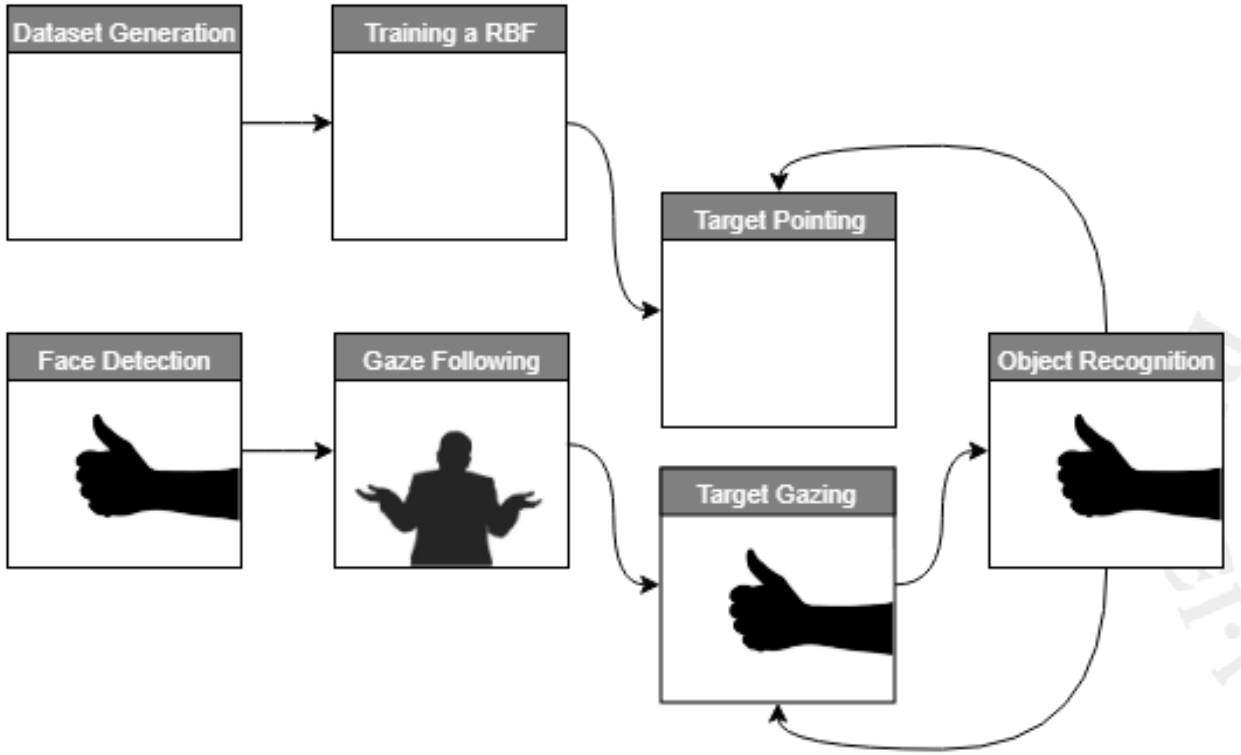
- Inspiration from Doniec, Sun and Scassellati[3][4]
- Motor babbling
- Learn how to move
  - Doing random movement
  - Observing the result
  - Many, many, many times
  - Network gets trained

Except this is developmental robotics

Pipeline


Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Pipeline


Dataset Generation
Training a RBF
Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Dataset generation

- Generate random motor commands

Dataset generation

- Generate random motor commands
- Object detection
  Note: we used the old object detection

Dataset generation

- Generate random motor commands
- Object detection
  Note: we used the old object detection
- If detected $\rightarrow$ store actuator settings and detected location.

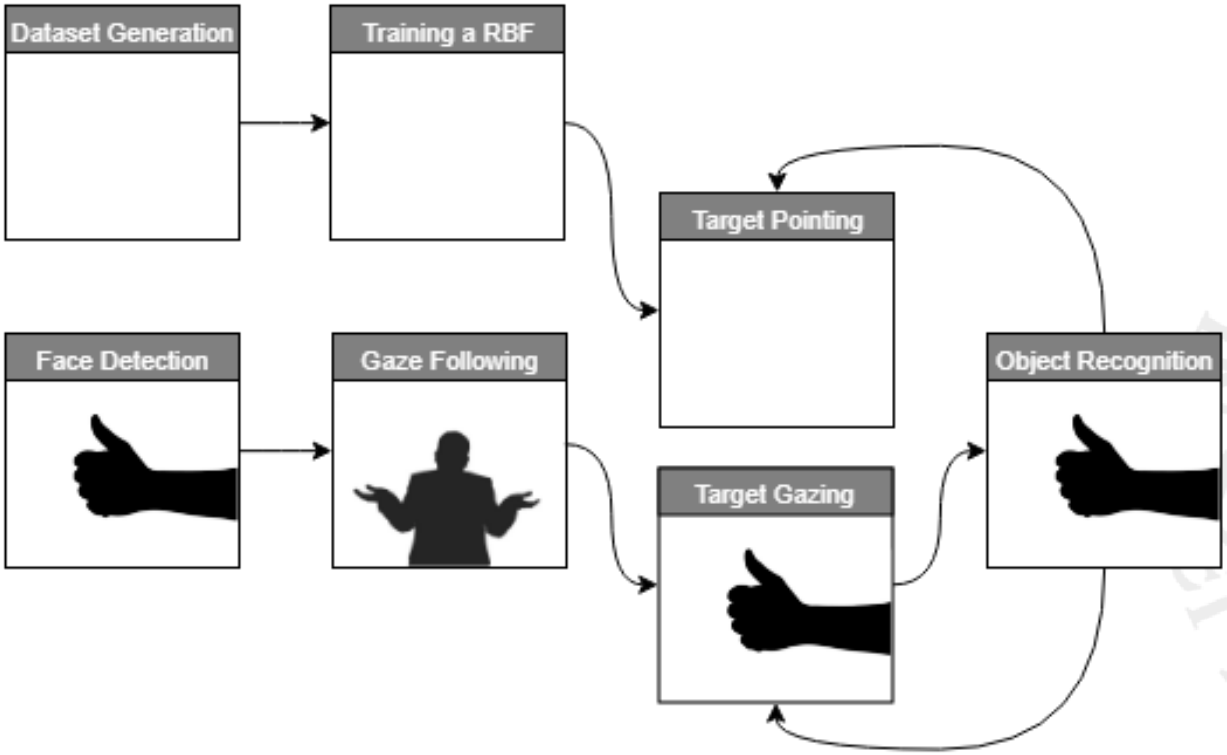
Dataset generation

- Generate random motor commands
- Object detection
  Note: we used the old object detection
- If detected $\rightarrow$ store actuator settings and detected location.
- Repeat

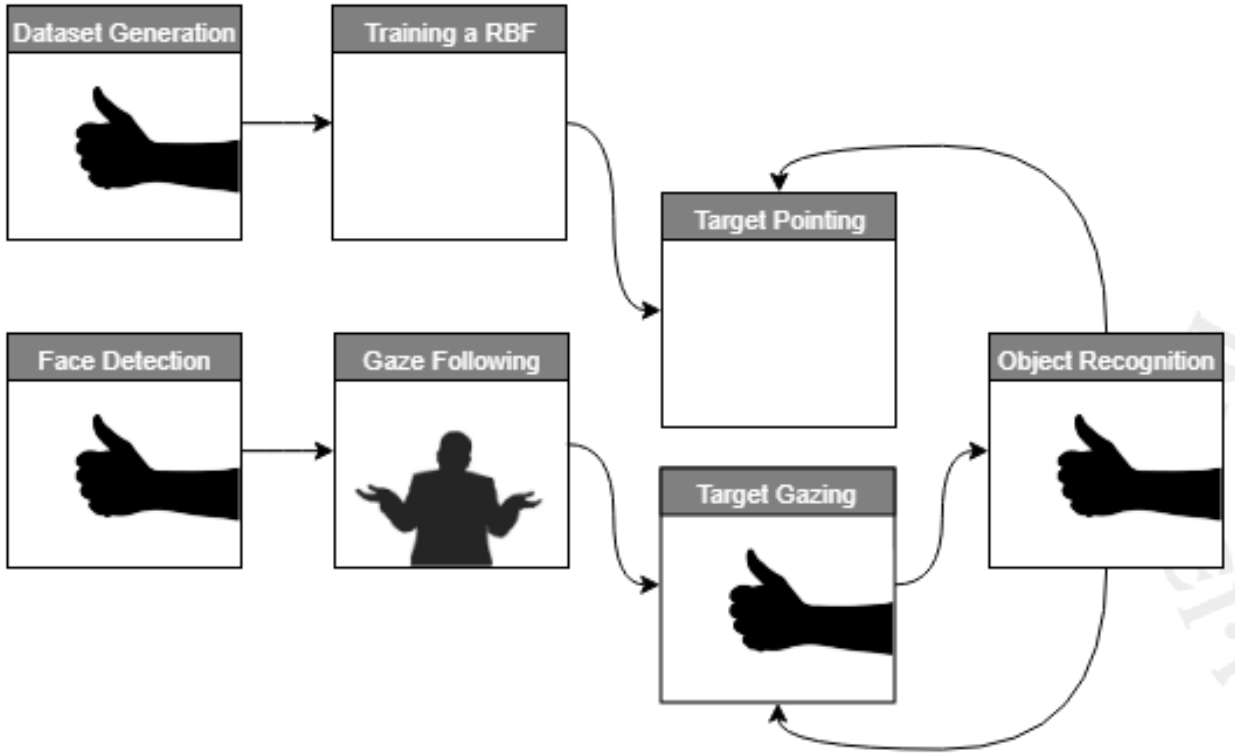
Dataset generation

- Generate random motor commands
- Object detection
  Note: we used the old object detection
- If detected $\rightarrow$ store actuator settings and detected location.
- Repeat

Pipeline


Dataset Generation
Training a RBF
Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Pipeline


Dataset Generation
Training a RBF
Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Training a RBF

- Implemented in Matlab

Training a RBF

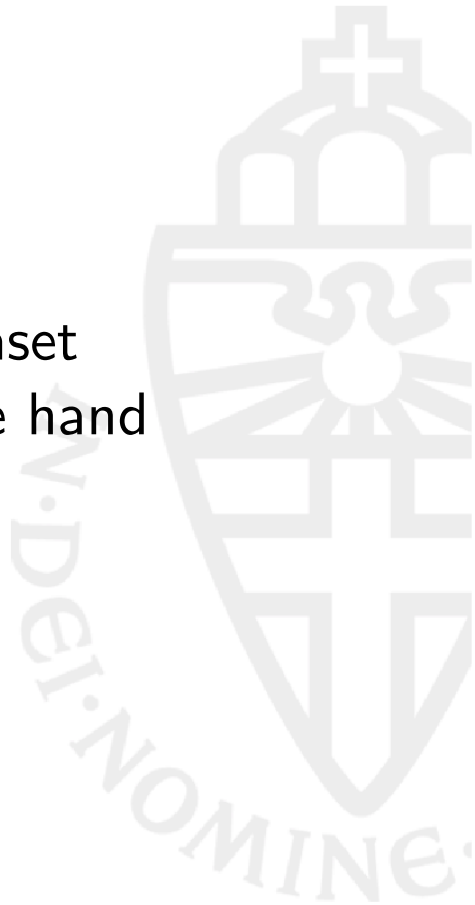
- Implemented in Matlab
- Trains the Radial Basis Function Network on dataset

Training a RBF

- Implemented in Matlab
- Trains the Radial Basis Function Network on dataset
  Input: head position and detected location of the hand

Training a RBF

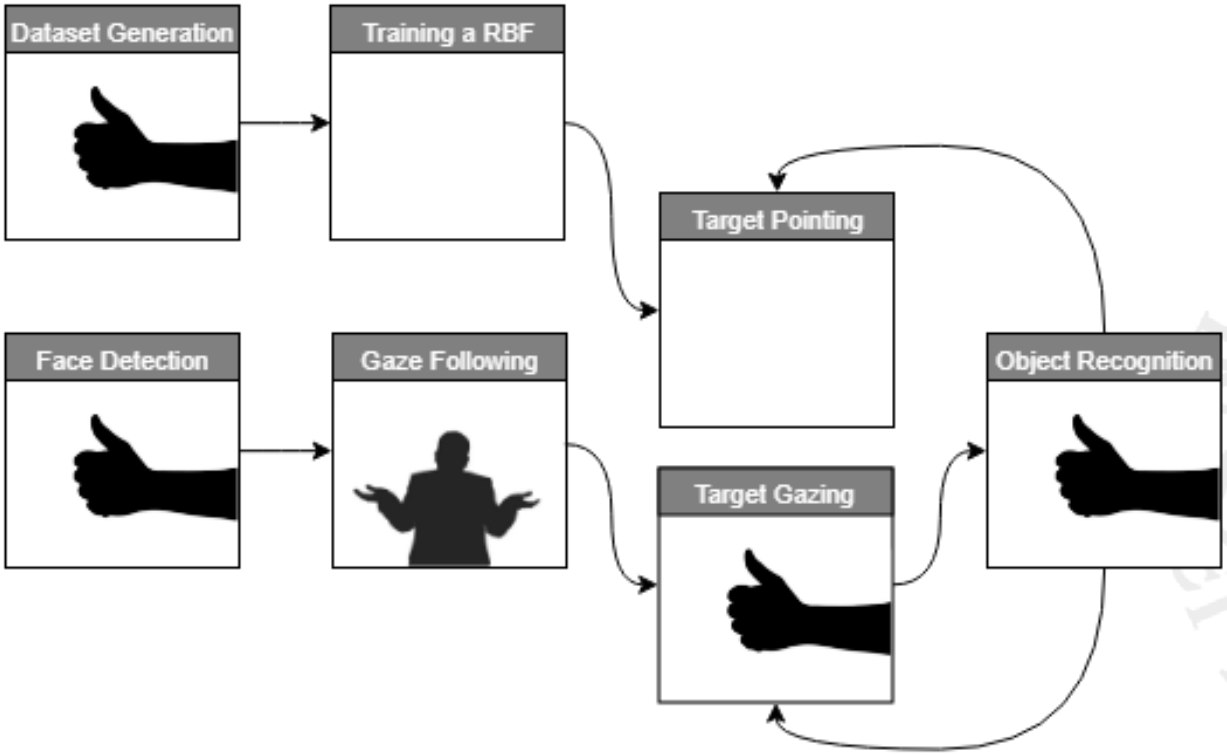
- Implemented in Matlab
- Trains the Radial Basis Function Network on dataset
  Input: head position and detected location of the hand
  Label: actuator settings of the arm



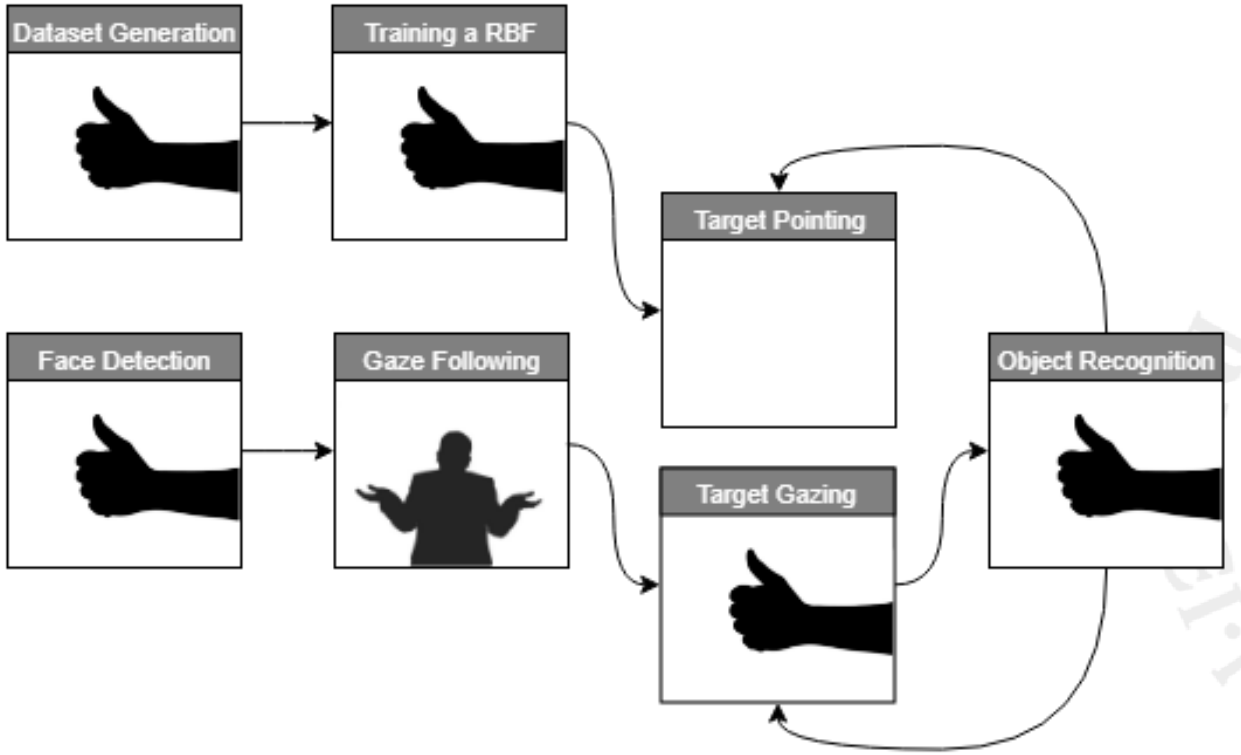
Training a RBF

- Implemented in Matlab
- Trains the Radial Basis Function Network on dataset
  Input: head position and detected location of the hand
  Label: actuator settings of the arm
- After training can be used to generate motor commands

Pipeline


Dataset Generation
Training a RBF
Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Pipeline


Dataset Generation
Training a RBF
Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Target pointing

- Starts matlab.engine

Target pointing

- Starts matlab.engine
- Feeds the RBF detected object location and head actuator settings

Target pointing

- Starts matlab.engine
- Feeds the RBF detected object location and head actuator settings
- RBF does it's magic

Target pointing

- Starts matlab.engine
- Feeds the RBF detected object location and head actuator settings
- RBF does it's magic
- Python receives motor commands

Target pointing

- Starts matlab.engine
- Feeds the RBF detected object location and head actuator settings
- RBF does it's magic
- Python receives motor commands
- Choose which arm to move

Target pointing

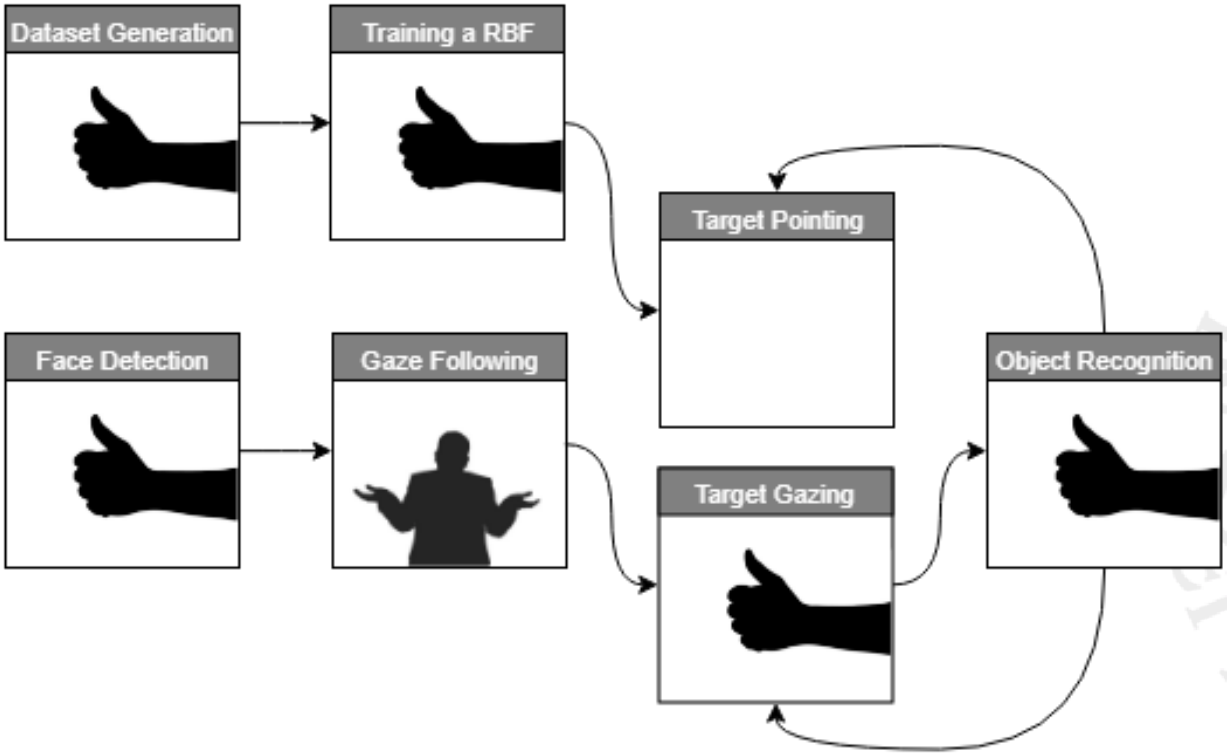
- Starts matlab.engine
- Feeds the RBF detected object location and head actuator settings
- RBF does it's magic
- Python receives motor commands
- Choose which arm to move
- Move arm according to the motor commands

Target pointing

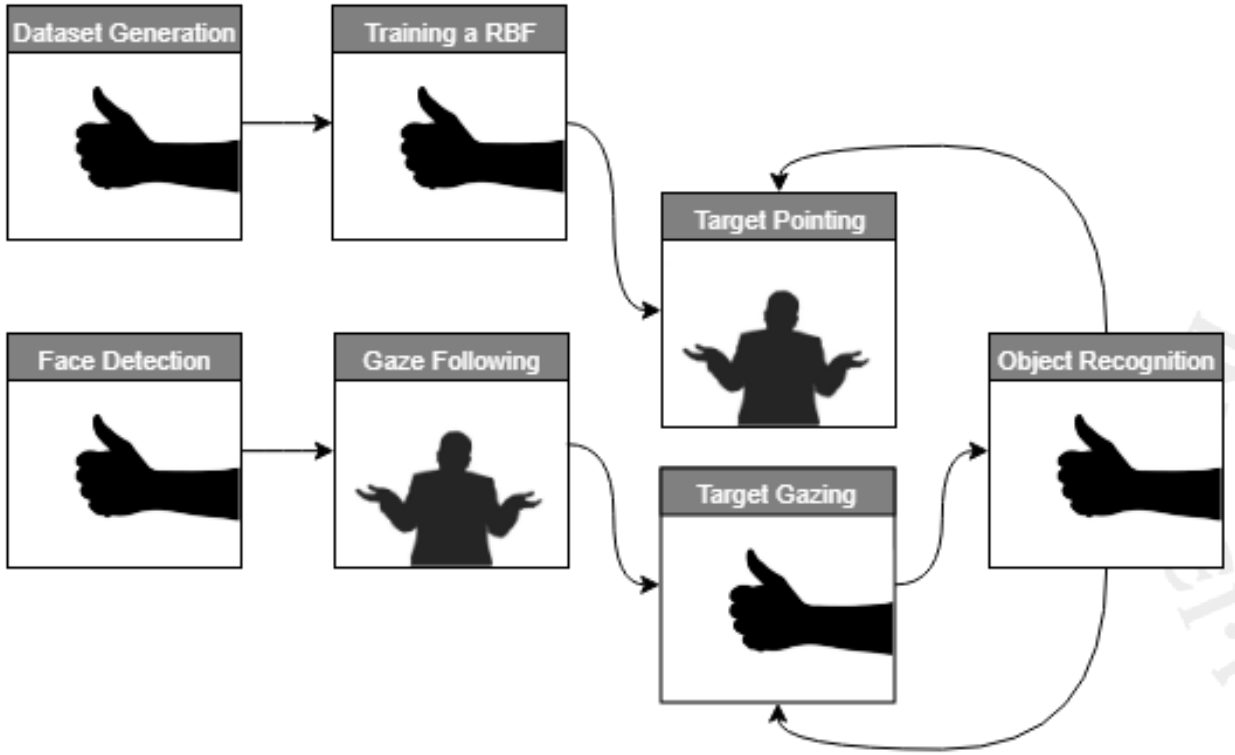
- Starts matlab.engine
- Feeds the RBF detected object location and head actuator settings
- RBF does it's magic
- Python receives motor commands
- Choose which arm to move
- Move arm according to the motor commands

Note: we manually set the elbow joint

Pipeline


Dataset Generation
Training a RBF
Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

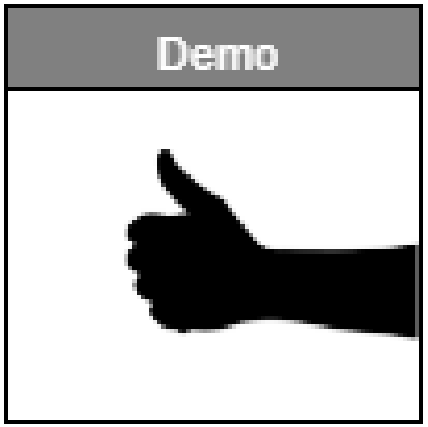
Pipeline


Dataset Generation
Training a RBF
Target Pointing
Face Detection
Gaze Following
Object Recognition
Target Gazing

Questions

Demo

Questions



Questions

Any Questions?

References

- P. Wolfert, *Gaze-following*, https://github.com/pieterwolfert/engagement-l2tor.git, 2018.
- A. Recasens*, A. Khosla*, C. Vondrick and A. Torralba, "Where are they looking?", in *Advances in Neural Information Processing Systems (NIPS)*, * indicates equal contribution, 2015.
- M. W. Doniec, G. Sun and B. Scassellati, "Active learning of joint attention", in *2006 6th IEEE-RAS International Conference on Humanoid Robots*, Dec. 2006, pp. 34–39. DOI: 10.1109/ICHR.2006.321360.
- G. Sun and B. Scassellati, "A fast and efficient model for learning to reach", *International Journal of Humanoid Robotics*, vol. 2, no. 04, pp. 391–413, 2005.