CS548 2015 Decision Trees / Random Forests

Showcase by: Lily Amadeo, Bir B Kafle, Suman Kumar Lama, Cody Olivier

Showcase work by Jamie Shotton, Andrew Fitzgibbon, Richard Moore, Mat Cook, Alex Kipman, Toby Sharp, Andrew Blake, Mark Finocchio
on
Real-Time Human Pose Recognition in Parts from Single Depth Images

Sources

Microsoft Kinect. (n.d.). Retrieved February 14, 2015, from http://hamlynkinect.wikispaces.com/Microsoft Kinect

Oberg, J., Eguro, K., Bittner, R., & Forin, A. (2012). Random Decision Tree Body Part Recognition Using FPGAs. *International Conference on Field Programmable Logic and Applications,* University of Oslo, Norway.

Oberg, J. (2011, October 7). The Kinect's Body Part Recognition Algorithm on an FPGA - Microsoft Research. Retrieved from http://research.microsoft.com/apps/video/dl.aspx?id=157648

Shotton, Jamie, Sharp, Toby, Kipman, Alex, Fitzgibbon, Andrew, Finocchio, Mark, Blake, Andrew, Cook, Mat & Moore, Richard (2013). Real-time Human Pose Recognition in Parts from Single Depth Images. Commun. ACM, 56, 116-124.

Xbox One. (n.d.). Retrieved from http://en.wikipedia.org/wiki/Xbox_One

Xbox 360. (n.d.). Retrieved from http://en.wikipedia.org/wiki/Xbox_360

Microsoft Kinect

- Video game console: XBox 360, XBox One
- Similar: Playstation, Nintendo Wii

- No game controllers/peripherals
- Use of “natural user interface”
- Features:
  - 3D motion capture
  - Facial recognition
  - Voice recognition

Microsoft Kinect

- Uses infrared laser light with speckle pattern

  - speckle effect : interference of many waves of same frequency, having different phases, amplitude.

- Tracks upto 6 people

  - 2 active players using motion analysis; <x, y, z>

- Automatic Sensor Calibration

  - Based on Gameplay

  - Based on physical Environment

Microsoft Kinect

- Vision based object recognition
- Pixel classification using Random Decision Trees (RDT)
- Algorithm: forest fire pixel classification algorithm
- Hardware: Field Programmable Gate Array (FPGA)
- Inferring body position:
  - Compute a depth map using structured light
    - Depth from focus
    - Depth from stereo
  - Machine learning

Decision Trees

Use of Decision Trees in Kinect
a) Efficiency
- computationally efficient

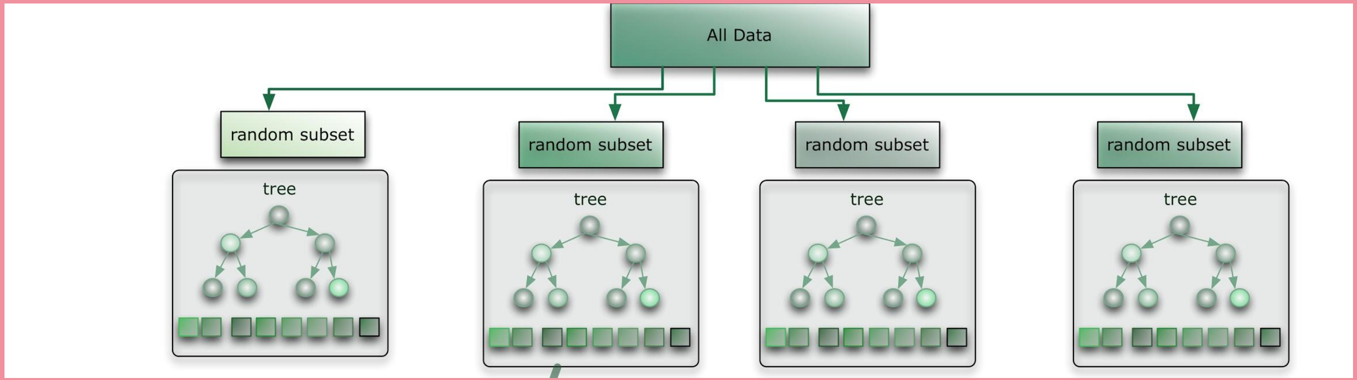
b) Relatively Easy to Update Algorithm
- Integrate new innovations
- Include new use cases

Random Forests

- Ensemble learning
  - Example: the Netflix prize
- Combined models
  - Trees, trees, more trees
- Bagging
  - Bootstrap aggregating
- Add more randomness
  - Feature bagging

Random Forest

- One tree trained on a subset of features
  - p features, sqrt p selected each time
- Another tree trained on a different subset of features



- Whole forest of trees

http://citizennet.com/blog/wp-content/uploads/2012/11/RF.jpg

Random Forest

Pros:

- Efficient
- Distributed
- Variable importance

Cons:

- Interpretability

Body Part Inference

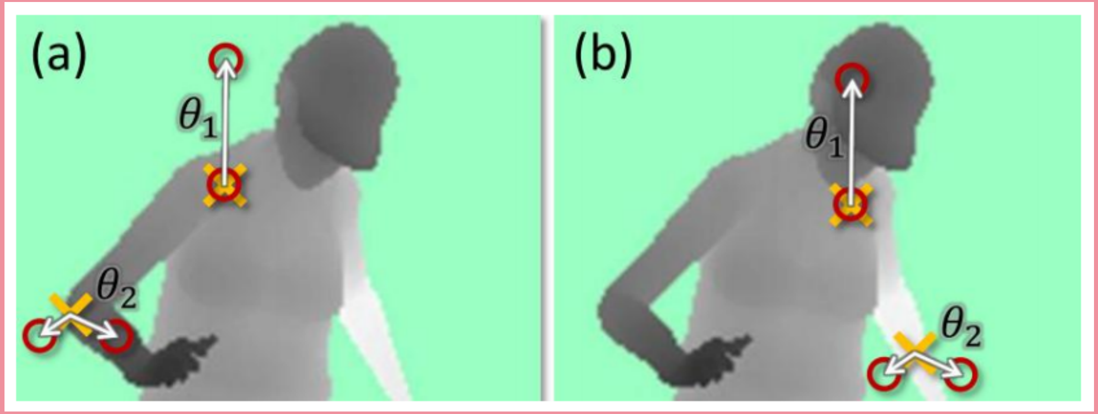
Define several body parts labels.

Parts could be changed to suit a particular application.

Small parts = accurately localized body joined.

Depth Image Features

- dI (x)= depth of pixel
- θ = offsets



$$f_\theta(I,\mathbf{x}) = d_I\left(\mathbf{x} + \frac{\mathbf{u}}{d_I(\mathbf{x})}\right) - d_I\left(\mathbf{x} + \frac{\mathbf{v}}{d_I(\mathbf{x})}\right)$$

$\frac{1}{d_I(\mathbf{x})}$ ensures features are depth invariant

Depth Feature continued

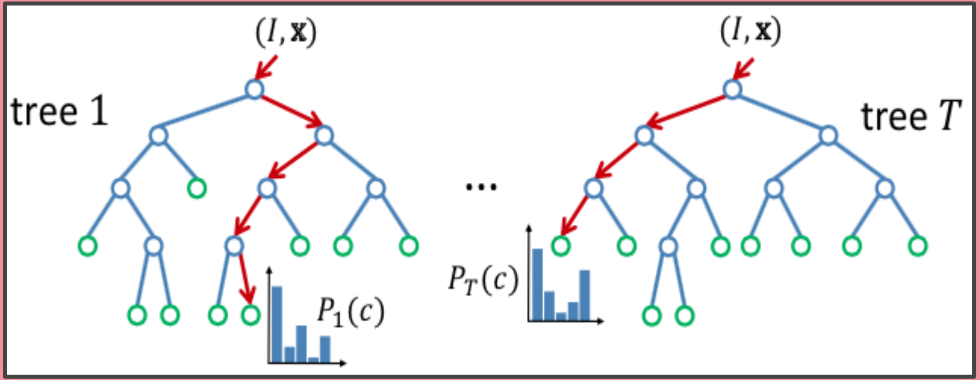
- $f_{\theta 1}$ looks upward : gives a large positive response at the upper portion of the body but close to zero near lower down the body
- Provide weak signal about which part of the body a pixel belongs to.
- Decision forest is what makes it accurate. Removes disambiguity.

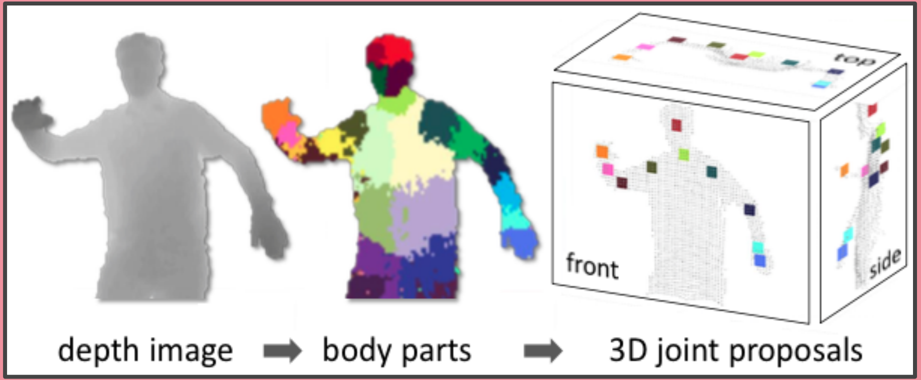
Decision Tree Creation

- Each tree gets: depth limit, a random 2000 pixel from each training image, set of candidate features
- Candidate features: parameters that determine how likely a pixel is a particular joint and a threshold
- Candidate features used in splitting subset in half
  - Pixels above and below threshold
- Entropy and Info Gain calculated from these two subset

Classification

- Probability distribution at leaves
- Distributions of trees in forest are averaged for classification of a pixel
- 31 joints calculated with mean-shift clustering





Experiment

- Forests: 3 trees, 20 nodes deep, 300k training images per tree, 2000 random pixels per image, 2000 candidate features, 50 candidate thresholds per feature
- Datasets:
  - 8808 real images, hand labeled
  - 5000 images synthesized from motion capture poses
  - Synthetic silhouette images

Results per Pixel

Figure 6. **Training parameters *vs.* classification accuracy.** (a) Number of training images. (b) Depth of trees. (c) Maximum probe offset.

Results for Joints

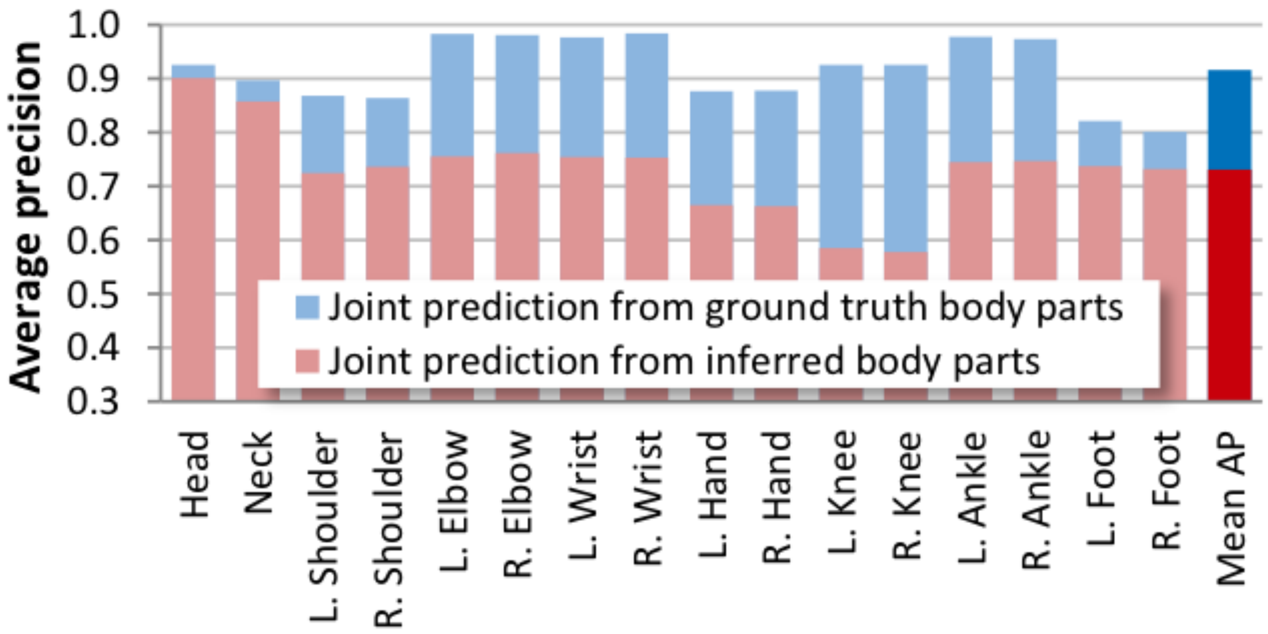


Figure 7. **Joint prediction accuracy.** We compare the actual performance of our system (red) with the best achievable result (blue) given the ground truth body part labels.

Conclusion

- Better accuracy than previous NN methods
  - Faster classification time than NN
- Better than Ganapathi et al. method
  - Doesn't exploit temporal and kinematic constraints