Randomized Trees for Human Pose Detection
(CVPR 2008)

Grégory Rogez, Jonathan Rihan, Srikumar Ramalingam, Carlos Orrite and Philip H. S. Torr

Computer Vision Lab – I3A
University of Zaragoza -SPAIN

Computer Vision Group
Oxford Brookes University - UK
Introduction: Goal

Full-body human pose detection
Introduction: Goal

Human detection
Introduction: Goal

Human pose recognition (2D and/or 3D)
Introduction: Goal

Full-body human pose detection
Introduction: Related Work

- **Human Detection**
  [Viola et al.‘03, Dalal & Triggs’05, Zhu et al.’06, Sabzmeydani & Mori’07, Gavrila’07, etc.]

- **Human Pose Recognition**
  [Shakhnarovich et al.‘03, Agarwal & Triggs’06, Mori & Malik’06, Thayananthan et al.’06, Rogez et al.’08, etc.]

- **Human Pose Detection**
  [Dimitrijevic et al.’06, Bissaco et al.’06]
Introduction: Related Work

Human Detection

• [Viola et al. ‘03, Viola et al. ‘05] Marr Prize 2003!!!
  → Integrate image intensity information with motion information & scan a detector over 2 consecutive frames of a video seq.

Input representation:

→ Use AdaBoost to select a subset of features

Haar filters:

→ Cascade architecture to make the detector efficient
Introduction: Related Work

Human Detection

• [Dalal & Triggs’05]
  → Use Histograms of Orientated Gradient (HOG)

→ Linear SVM to weight each cell of each block and classify
Introduction: Related Work

Human Detection

- [Dalal & Triggs’05]

The most **important cells** are the ones that typically contain major human contours (especially the **head and shoulders** and the **feet**)

→ Demonstrate how **HOGs outperform** existing feature sets.
**Introduction: Related Work**

**Human Detection**

- [Zhu et al.’06]
  
  → Integrate the **cascade-of-rejectors** approach with **HOG** features to achieve a fast and accurate human detector
  
  → Use **AdaBoost** for feature selection
  
  → Compute the separating hyperplane using a linear **SVM**.

- first four levels in the cascade only contain four SVM classifiers each, and reject about 90% of the detection windows.

- the average number of blocks to be evaluated for each detection window is as low as 4.6.
Introduction: Related Work

Human Detection

• [Sabzmeydani & Mori’07]
  → Introduce an algorithm for learning Shapelet features
  → Use AdaBoost 1\textsuperscript{st} to create these shapelets as a combination of oriented gradient responses, and 2\textsuperscript{nd} to select a subset of them.
Introduction: Related Work

Human Detection

- [Gavrila’07] → Bayesian hierarchical shape matching
Introduction: Related Work

Human Detection

• [Gavrila’07] → Bayesian hierarchical shape matching

Coarse to fine grid as search goes from top to intermediate and leaf level

Intermediate matching results for a 3-level template tree
Introduction: Related Work

Human Pose Recognition

• [Shakhnarovich et al.’03]
  → **Exemplar based** approach
  → Parameter Sensitive Hashing (PSH)
  → Orientated gradient (similar to HOG)

The main idea of PSH is to find a feature space in which the similarity in terms of L1 distance would be closely related to similarity in parameter (pose) space.
Introduction: Related Work

Human Pose Recognition

• [Shakhnarovich et al.’03]

Figure 4. Examples of upper body pose estimation (Section 4). Top row: input images. Middle row: top PSH match. Bottom row: robust constant LWR estimate based on 12 NN. Note that the images in the bottom row are not in the training database - these are rendered only to illustrate the pose estimate obtained by LWR.
Introduction: Related Work

Human Pose Recognition

• [Mori & Malik’06]
  → Shape context matching to each stored exemplar
  → The locations of the body joints are finally transferred from the exemplar view that best matched the input image, to this test shape.
Introduction: Related Work

Human Pose Recognition

- [Agarwal & Triggs'06]
  - Shape Context of silhouette
  - Select relevant features using RVM regression
  - Pose estimation formulated as a mapping from feature space \( z \) to pose space \( x \) using a mixture of Regressor on the joint density \((z, x)\)

\[
\begin{pmatrix} z \\ x \end{pmatrix} \sim \sum_{k=1}^{K} \pi_k \mathcal{N}(\mu_k, \Gamma_k)
\]
Detection + Pose Recognition

• [Dimitrijevic et al.’06]
  → template-based approach to detecting specific walking pose
  → estimate and store the relevance of the silhouette parts
  → convert Chamfer distance to meaningful probability estimates
Introduction: Related Work

Detection + Pose Recognition

- [Dimitrijevic et al.’06]
  → template-based approach to detecting specific walking pose
Detection + Pose Recognition

- [Bissaco et al.'06]
  - Is there a human in the image? and, if so, 2) what is a low-dimensional representation of the pose?
  - Use Latent Dirichlet Allocation (LDA) model to represent the statistics of the gradient orientation features.
  - Generative probabilistic model which allows for automatic discovery of pose information.

Detection rate similar to [Dalal&Triggs '05] but with a low-dimensionnal representation of the pose
• Detect detailed poses (joints)
  Previous works do not provide estimation of joint location…

• Exemplar-based approach as Classification problem
  Nearest neighbour search…

• No assumption of available segmentation
  Moving camera seq. & single image classification…

• Use HOG features
  Good results in human detection & pose recognition…

• Focus on walking action
  But propose solutions that can be generalized…

• Use HumanEVA I for training and HumanEVA II for testing
  The method will be easy to compare with…
Introduction: Overview of the Approach

Training Images

Off-line

Images Alignment
Random Forest Generation
Class Definition
Regressor Learning
3D and 2D Training Poses

On-line

Input Image
Trees
Bounding Box, Class $C_i$

Human Detection & Classification
Pose Estimation

2D joints location
3D joints angles

Random Forest Generation
Class Definition
Random Forest Generation
Class Definition
1. Introduction
2. Pre-processing Steps
   • Alignment of training data
   • Class definition
3. Random Forest Generation
   • Selection of discriminative features
   • Bottom-up hierarchical tree learning
   • Random selection of features
4. Human Pose Detection
5. Experiments
6. Conclusions and Discussions
Introduction

Pre-processing Steps

Random Forest Generation

Human Pose Detection

Experiments

Conclusions

**Pre-processing Steps: Alignment**

- Establish correspondences between training images
  - Features selection much easier

- Other approaches:
  - Manual process for images alignment

INRIA & MIT Databases [Dalal&Triggs’05, Zhu et al.’06, Sabzmeydani&Mori’07]

Manually marked boxes [Viola et al. ‘03, Viola et al. ‘05]
• Other approaches:
  → Clean **synthetic** training silhouettes
Pre-processing Steps: Alignment

- Other approaches:
  → Manually labelled training shapes

[Gavrila’07]

[Roge et al.’08]
Pre-processing Steps: Alignment

- 3D-2D projection of the joints
- 2D Poses alignment
- Rescale and center in a 96x160 bounding-box
Pre-processing Steps: Alignment

For each training image:
- Compute the transformation $T$ between original 2D joints locations and normalized ones
- Apply $T$ to the image

Average gradient image over HumanEVA training examples
Pre-processing Steps: Class definition

• Similar 3D poses can have very different appearance depending on the viewpoint

→ PSH seems difficult to apply with extensive viewpoint changes since they use the distance in pose space to define the hash function in the feature space.

→ We need to include the viewpoint information into the class definition
Pre-processing Steps: Class definition

- Use a 2D manifold where viewpoint & action are 1D manifolds → because of cyclicity of viewpoint & walking we obtain a torus-manifold

- Align the gait sequences temporally [Urtasun’05] and map them to a torus manifold [ElGammal’06]

- Define the classes on this torus by discretizing viewpoint and gait cycle [Rogez’08]
  → in this paper we define 16x12 = 192 classes
Pre-processing Steps: Class definition
Discriminative Features

- Adaboost and SVM are usually used to select useful features.

**Intuition**

Pre-processing Steps
Random Forest Generation
Human Pose Detection
Experiments
Conclusions

**Discriminative Features**

- Weighted Hogs from SVM [Dalal & Triggs’05]
- Best Hog & Haar filter from Adaboost [Zhu et al.’06]
- Shapelets from Adaboost [Sabzmeydani & Mori’07]

→ very time consuming but the result is already there!!
• New method to select most informative HOG blocks and favour locations that we expect to be more discriminative.

• Log-likelihood ratio for the $i^{th}$ class:

$$L_i = \log\left( \frac{p(E, C_i)}{p(E)} \right)$$

Average gradient image for Classes 1 to N.  Average gradient image for Classes i
Discriminative Features

Log-likelihood ratio of each class and position on the torus manifold
• Random HOG Block Sampling proportional to Log-likelihood
Bottom-up Hierarchical Tree Learning
Random Selection of Useful Features

**Algorithm 1: Discriminative Features Selection**

- **input**: Hierarchical structure and training images.
- **output**: List of discriminative HOG blocks.

```plaintext
for each level l do
  for each node n do
    Compute edge probability $p(E_{l,n})$ over images that pass through $n$;
    for each child c do
      Compute edge probability $p(E_{l,n,c})$ over images that pass through $c$;
      Compute log-likelihood $L_{l,n,c}$ (cf Sec. 3.1);
      Sample $n_{h_i}$ HOG blocks $\{h_{i1}^{n_{h_i}}\}$ from $L_{l,n,c}$;
      for each $h_i$ do
        Extract $h_i$ for all the images reaching $n$;
        Compute the mean histogram $\bar{h}_i$ over images that pass through $c$;
        Compute $L_2$ distances to $\bar{h}_i$;
        Compute the best threshold $t_i$ that splits the data and minimizes FP and FN rates;
      end for
    end for
  end for
end for
```

**Node n**

**Child c**

$H_{n,c,1}, \ldots, H_{n,c,N}$
Random Selection of Useful Features

- Grow an ensemble of trees, a **forest**, by **randomly** choosing one of the N selected HOG Blocks for each branch of the tree.

Tree 1 --- Tree t ------ Tree 200

\[ H_{n,c,j} \]

Node n

\[ H_{n,c,1}, \ldots H_{n,c,j}, \ldots H_{n,c,N} \]
• Given an input 96x160 image, each tree gives a binary decision for each class

• It results in a distribution over all classes when considering the forest
Human Pose Detection

Introduction
Pre-processing Steps
Random Forest Generation
Human Pose Detection
Experiments
Conclusions

On-line
Human Detection & Classification
Bounding Box, Class C,

Input image
Experiments
Human Pose Detection

Introduction
Pre-processing Steps
Random Forest Generation
Human Pose Detection
Experiments
Conclusions
Experiments

<table>
<thead>
<tr>
<th>Subject</th>
<th>Camera</th>
<th>Frames</th>
<th>Mean (Std) from [17]</th>
<th>Mean (Std) using RT</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2</td>
<td>C1</td>
<td>1-350</td>
<td>16.96 (4.83)</td>
<td>12.98 (3.5)</td>
</tr>
<tr>
<td>S2</td>
<td>C2</td>
<td>1-350</td>
<td>18.53 (5.97)</td>
<td>14.18 (4.38)</td>
</tr>
</tbody>
</table>
Conclusions

• Novel approach for exemplar-based human pose Detection

• Does not require silhouette segmentation
  → can be applied to moving camera sequences and single images...

• our Random Forest allows to model distribution over poses