Face Detection and Tracking from Video Imagery

Qiang Ji, Peng Wang, Andrew Janowczyk, and Ted Carlson

Department of Electrical, Computer and System Eng. Rensselaer Polytechnic Institute (RPI)
jiq@rpi.edu
Project Goals

• The goal of this project is to develop improved algorithms for face detection and tracking from video imagery for use in the facial recognition system.

• The primary application of our system is for security monitoring and surveillance in public places, where the number of people vary dynamically, face poses and scales vary, and faces are small (20x20 pixels) and are of low resolution.
System Components

- **People Detection and Tracking**
- **Face Detection**
- **Integrated Face Tracking**

Motion detection and tracking based on motion estimation and Particle Filtering

Appearance-based face detection from detected human

Face Tracking
System Component I:
People Detection and Tracking

Two probabilistic methods are applied:

- People detection based on motion estimation and Bayesian propagation over frames
- People tracking based on particle filtering
People Detection

We apply a people detection method based on robust optical flow estimation, background modeling, prior motion modeling using MRF, and Bayesian motion propagation over frames.
Robust Optical Flow Estimation

- **Feature**: optical flow vector \( \mathbf{v}(x, y) = (v_x, v_y) \) from robust optical flow estimation
  - LMS followed by LS to solve local image derivatives;
  - Globally minimize estimation error;

Robust motion estimation
Background Motion Model

- Two states for motion detection:
  - \( H^0 \) : Background
  - \( H^1 \) : Moving Objects (People)

- Model background motion with a dominant background motion with \( v^0(x, y) \) plus some additive Gaussian noise.

\[
P(v(x, y) | H^0) \sim N(v^0(x, y), \sigma^2)
\]

Learning model parameters from motion vectors
Derive Prior Distribution through Markov Random Field

- Model prior distribution through Markov Random Field (MRF)

\[
P(x, y) = \frac{1}{Z} e^{-\frac{U(x, y)}{T}} \quad U(x, y) = \sum_{c \in C} D_c (x, y)
\]

- The prior probability of a point only depends on the configuration of its neighbors.
People Detection

- Derive human motion likelihood model from background motion model and prior distributions

\[
P(v | H^1) = \frac{P(v) - P(v | H^0)P(H^0)}{P(H^1)}
\]

- People detection based on Bayesian classifier

\[
\begin{align*}
H^0 : \text{background, } & \text{ if } P(H^0 | v) > P(H^1 | v) \\
H^1 : \text{people, } & \text{ if } P(H^1 | v) > P(H^0 | v)
\end{align*}
\]
Bayesian Propagation to Improve People Detection

- Motion detection at only one frame has some false and missing detections, we refine motion detection results over frames by propagating posterior probabilities.

- Bayesian rule over frames

\[
P(H_t \mid V_{1:t}) \propto P(v_t \mid H_t) \sum_{H_{t-1}} P(H_t \mid H_{t-1}) P(H_{t-1} \mid V_{1:t-1})
\]

- \( V_{1:t} = (v_1, \ldots, v_t) \) is feature history until time t

- Posteriori probability at frame t \( P(H_t \mid v_t) \) is estimated from:
  - Posterior probability at previous frames \( t-1 \) \( P(H_{t-1} \mid v_{t-1}) \)
  - the transition probability \( P(H_t \mid H_{t-1}) \)
  - the likelihood model \( P(v_t \mid H_t) \)
State Transition Model

- Assuming constant object intensity with Gaussian noise, we model transition probability by local intensity changes at position \((x,y)\).

\[
p = \exp\left\{ \frac{(I_{t+1}(x + v_x, y + v_y) - I_t(x, y))^2}{2\sigma^2} \right\}
\]

\[
\begin{pmatrix}
P(H_t^0 | H_{t-1}^0) & P(H_t^1 | H_{t-1}^0) \\
P(H_t^0 | H_{t-1}^1) & P(H_t^1 | H_{t-1}^1)
\end{pmatrix} = \begin{pmatrix}
p & 1-p \\
1-p & p
\end{pmatrix}
\]
People detection results are refined frame by frame. The false detections caused by illumination change and random noise are eliminated by propagation over frames.
Motion Tracking with Particle Filtering (PF)

The goal is to efficiently track the detected people from frame to frame through a recursive propagation of probability distribution. PF allows to track multiple and unknown number of targets.

- State vector: \( S_t = (x, y, v_x, v_y) \)
- Observation vector: \( z_t = (x, y) \)
- Observations up to t: \( Z_{1:t} = (z_1, z_2, \ldots, z_t) \)
Motion Tracking with Particle Filtering (cont’d)

We approximate $p(s_t | Z_{1:t})$ using a cloud of particles:

$$\{S_t^{(i)}\}_{i=1}^N$$

With the associated weights:

$$\{W_t^{(i)}\}_{i=1}^N$$

Particles are initially obtained near the detected human bodies. The weights are determined based on their distances to the detected bodies.
Human Motion Model

- Assumption:
  comfortable speed with constant acceleration

- Five tracker models are studied

<table>
<thead>
<tr>
<th>Tracker</th>
<th>State Vector</th>
<th>Observation Vector</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>((x, y))</td>
<td>((x, y))</td>
</tr>
<tr>
<td>2</td>
<td>((x, y, v_x, v_y))</td>
<td>((x, y))</td>
</tr>
<tr>
<td>3</td>
<td>((x, y, v_x, v_y, a_x, a_y))</td>
<td>((x, y))</td>
</tr>
<tr>
<td>4</td>
<td>((x, y, v_x, v_y))</td>
<td>((x, y, v_x, v_y))</td>
</tr>
<tr>
<td>5</td>
<td>((x, y, v_x, v_y, a_x, a_y))</td>
<td>((x, y, v_x, v_y))</td>
</tr>
</tbody>
</table>
Motion Tracking Models

- State transition model $p(s_{t+1} | s_t)$

$$
\begin{bmatrix}
  x \\
  y \\
  v_x \\
  v_y
\end{bmatrix}_t =
\begin{bmatrix}
  1 & 0 & \Delta t & 0 \\
  0 & 1 & 0 & \Delta t \\
  0 & 0 & 1 & 0 \\
  0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
  x \\
  y \\
  v_x \\
  v_y
\end{bmatrix}_{t-1} +
\begin{bmatrix}
  w_x \\
  w_y \\
  w_{vx} \\
  w_{vy}
\end{bmatrix}
$$

- $p(z_t | s_t)$, the observation likelihood distribution returns the likelihood of the state $s_t$, given the observation $z_t$.

- $p(s_t)$ prior pdf gives the initial distribution of the system states from previous time frame.
Probability Propagation

Prediction

\[ p(s_t \mid Z_{1:t-1}) = \sum p(s_t \mid s_{t-1}) p(s_{t-1} \mid Z_{1:t-1}) \]

Observation Likelihood \( p(z_t \mid s_t) \) based on the background modeling

Posterior probability

\[ p(s_t \mid Z_{1:t}) = \frac{p(z_t \mid s_t)p(s_t \mid Z_{1:t-1})}{\sum_{i=1}^{N_{t-1}} p(z_t \mid s_t)p(s_t \mid Z_{1:t-1})} \]
PDF propagation
Roaming Novelty Detector

- Scan the image dynamically to efficiently detect previously missed people

Series of roaming novelty windows
Demo: Indoor Sequence
System Component II: Face (head) Detection

Multi-view face detection with an ensemble of boosted SVMs

- Produce a database of faces from real world scenes
- Face detection by boosting and bagging SVMs
Current Face Database

1. Mostly captured with artificial illumination and poses
2. Mostly frontal faces and contain only faces (not head)
3. Face is captured in relatively short distance with rich face expression, unsuitable for face detection for surveillance applications.
Multi-View Face Database

We produce a face database with multiple views under real complex scenarios.

- Above 2000 labeled faces captured under real scene in public places.
- With multiple poses and scales, including frontal faces and profile faces.
- Available at
  http://www.ecse.rpi.edu/~cvrl/face_project/database/data_base.htm
Face Detection with an Ensemble of Boosted SVMs

- Pattern to be tested
- Preprocessing
- Face data (frontal and profile faces)
- Boosted SVMs
- Nonface data

Bagging Ensembles

- Majority Voting
- Post processing
- Face
- Non face
Train Hierarchical SVMs by Boosting

- Boosting: repeatedly train the SVM classifiers with the hard-to-classify samples to form boosted SVMs.

- Cascading the boosted SVMs for efficient course-to-fine classification.

\[ \text{SVM 1} \rightarrow \text{SVM 2} \rightarrow \text{SVM 3} \]
Bagging SVMs to Eliminate False Detections

- Bagging: average multiple classifiers \( f_i(x), i = 1, \ldots, N \) to obtain robust results.

\[
f(x) = \frac{1}{N} \sum_{i=1}^{N} f_i(x)
\]

- Bagging SVMs can eliminate the false detections that are difficult to handle by a single SVM.

<table>
<thead>
<tr>
<th>SVM</th>
<th>Polynomial (second order)</th>
<th>Polynomial (third order)</th>
<th>RBF</th>
<th>Bagging</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of SVs</td>
<td>870</td>
<td>862</td>
<td>784</td>
<td>N/A</td>
</tr>
<tr>
<td>Detection Rate</td>
<td>90.7%</td>
<td>89.5%</td>
<td>91.6%</td>
<td>92.5%</td>
</tr>
<tr>
<td>False Rate</td>
<td>1.59%</td>
<td>2.19%</td>
<td>1.40%</td>
<td>0.85%</td>
</tr>
</tbody>
</table>

Table 1. Test performance of bagging SVMs
Some Detection Results

Average detection rate of 93%
System Component III: Face tracking with particle filtering

- Particles are initialized around the detected faces
- Particle state include: face position, size, and velocity
- Set detected face as template for subsequent measurement
- First-order kinematics motion model to capture face dynamics
Face Model Updating

- In sequences, face appearance change due to:
  - Face pose change during moving;
  - Scale change;
  - Illumination condition change.

- Not necessary to perform face detection at each frame. Only activate face detector to correct tracking results when face appearance changes significantly, or face tracking get lost.
When to Activate Face Detector?

- Face detection and tracking error is reduced when particle filtering converges.

- Entropy criteria is proposed to measure the convergence of the particle filter.

\[ Entropy = - \sum_{i} P_i \log P_i \]

\( P_i \) is the posterior probability of \( i \)th particle.

- If \( Entropy > \) threshold, then activate face detector to relocate face.
Face Pose Determination

Tracked faces are labeled as frontal (red) or profile face (green), based on the output of the face detector. Frontal faces are collected for subsequent face recognition.

Red rectangle indicates useful poses for recognition
Demo: Tracking Results
System Application

Surveillance statistics can be obtained from our face detection and tracking methods.

- Statistics: the number of people passing by, the duration they stay in the area, their moving speed and direction.

- Face multifold: extract multiple faces of different views of people passing by in video for face recognition
Extract Multiple Faces from Video for Recognition

More faces extracted from video

1. 
2. 
3. 
4.
Conclusion

1. Develop improved algorithms for
   • Detection and tracking multiple people in a dynamic environment
   • Detection and tracking faces of different poses and scales in real world and complex scenarios

2. Construct a database of over 2000 faces of different poses and scales, suitable for face detection and tracking in surveillance applications
Future Work

- Further integrate different components
- Improve speed, accuracy, and robustness
- Explore the possibility of face detection and tracking using multiple cameras via active sensory fusion