List of algorithms iii

16 Motion analysis 1

16.1 Differential motion analysis methods 9
16.2 Optical flow 16
  16.2.1 Optical flow computation 17
  16.2.2 Global and local optical flow estimation 24
  16.2.3 Combined local–global optical flow estimation 26
  16.2.4 Optical flow in motion analysis 30
16.3 Analysis based on correspondence of interest points 38
  16.3.1 Detection of interest points 39
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>16.3.2 Correspondence of interest points</td>
<td>40</td>
</tr>
<tr>
<td>16.4 Detection of specific motion patterns</td>
<td>46</td>
</tr>
<tr>
<td>16.5 Video tracking</td>
<td>61</td>
</tr>
<tr>
<td>16.5.1 Background modeling</td>
<td>62</td>
</tr>
<tr>
<td>16.5.2 Kernel-based tracking</td>
<td>72</td>
</tr>
<tr>
<td>16.5.3 Object path analysis</td>
<td>83</td>
</tr>
<tr>
<td>16.6 Motion models to aid tracking</td>
<td>89</td>
</tr>
<tr>
<td>16.6.1 Kalman filters</td>
<td>91</td>
</tr>
<tr>
<td>16.6.2 Particle filters</td>
<td>101</td>
</tr>
<tr>
<td>16.7 Summary</td>
<td>119</td>
</tr>
</tbody>
</table>
Adaboost Approach to Detection of Motion Patterns

- image-based and motion-based information are used simultaneously
- detection of short-term motion patterns rather than on tracking over extended periods of time
- closely related to the Adaboost object detection

- pedestrian motion
- small set of simple rectangle filters trained on a set of examples
- multiscale
- filters work with short temporal image sequences
• motion detected as temporal differences in corresponding image blocks
• size of blocks — analysis scale
• blocks of different sizes

• computational efficiency
• motion direction derived from differences between shifted image blocks
• shift by \( \psi \) pixels – defined with respect to detection scale
• image frame acquired at time \( t, t + \delta t \)
• Five highly relevant

\[
\Delta = \text{abs}(I_t - I_{t+1}), \\
U = \text{abs}(I_t - I_{t+\delta t} \uparrow), \\
D = \text{abs}(I_t - I_{t+\delta t} \downarrow), \\
L = \text{abs}(I_t - I_{t+\delta t} \leftarrow), \\
R = \text{abs}(I_t - I_{t+\delta t} \rightarrow),
\]  

(16.37)

Figure 16.12: Motion and appearance difference images derived according to equation (16.37). Image $R$ has the lowest energy and as such, corresponds to the right-to-left direction of motion.
Filters $f_k$ measure magnitude of motion

$$f_k = r_k(S)$$ \hspace{1cm} (16.38)

Several filter types ...

$$f_i = r_i(\Delta) - r_i(S)$$ \hspace{1cm} (16.39)

... likelihood that region is moving in a tested direction $\uparrow, \downarrow, \leftarrow, \text{ or } \rightarrow$

$S$ is one of the difference images $\{U, D, L, R\}$

$r_i$ is a single rectangle sum within the detection window.
Motion shear can be determined using filters

\[ f_j = \phi_j(S) \]  \hspace{1cm} (16.40)

Filters \( f_m \) ... detecting image patterns of expected static image properties

\[ f_m = \phi(I_t) \]  \hspace{1cm} (16.41)

- filters \( f \) – from integral image
- filters \( f \) can be of any size, aspect ratio, or position (as long as they fit in image block)
- large number of filters
- best subset ... to separate moving objects with motion-specific properties from the rest of the image
16.4 Detection of specific motion patterns

- Classifier $C$ – linear combination of selected features
- after AdaBoost training phase – thresholded sum of features

\[
C(I_t, I_{t+\delta t}) = 1 \quad \text{if} \quad \sum_{s=1}^{N} F_s(I_t, I_{t+\delta t}) > \theta, \\
= 0 \quad \text{otherwise.}
\] (16.42)
16.4 Detection of specific motion patterns

- Feature $F_s$ – thresholded image

\[ F_s(I_t, I_{t+\delta t}) = \alpha \quad \text{if} \quad f_s(I_t, I_{t+\delta t}, \Delta, U, D, L, R) > t_s, \]
\[ = \beta \quad \text{otherwise}, \]  \hspace{1cm} (16.43)

- $t_s \in \mathcal{R}$ is a feature threshold
- $f_s$ is one of filters $f$

- $N$ features $f_s$ are selected using AdaBoost process from all considered filters
- these filters are a function of one or more parameters $I_t, I_{t+\delta t}, \Delta, U, D, L, \text{and/or } R$
• $\alpha$, $\beta$, $t_s$, and $\theta$ – computed during the AdaBoost training process

• each of $N$ rounds chooses from the full set of motion and appearance features

• $\rightarrow$ a mix of features balancing the appearance and motion descriptors is selected
• motion-invariant detection of object motion speed is achieved via different shifts $\psi$
• obtained during training – scaling all training samples to a pre-determined base resolution (i.e., bounding block size with respect to the pixel counts in the $x$ and $y$ directions)
• e.g., base resolution of $20 \times 15$ pixels was used by Viola/Jones
• multi-scale behavior achieved by operating on image pyramids

\[
\begin{align*}
\Delta^l &= \text{abs}(I^l_t - I^l_{t+1}) , \\
U^l &= \text{abs}(I^l_t - I^l_{t+\delta t \uparrow}) , \\
D^l &= \text{abs}(I^l_t - I^l_{t+\delta t \downarrow}) , \\
L^l &= \text{abs}(I^l_t - I^l_{t+\delta t \leftarrow}) , \\
R^l &= \text{abs}(I^l_t - I^l_{t+\delta t \rightarrow}) ,
\end{align*}
\]  

(16.44)
• $l$ ... pyramid level

• features computed from the pyramidal representations in a scale-invariant fashion
• scale factor of 0.8 for successive pyramid levels shown to work – all the way down to the pre-determined size of the base-resolution image block ($20 \times 15$ pixels in the discussed case).
• Once features selected, a boosted cascade of classifiers

• Simple classifiers with high detection rates and relatively high false positive rates are employed in early stages

• More complex classifiers using larger numbers of features are used in the later cascade stages

• Each stage of the cascade attempts to reduce both the detection and the false positive rates

• → goal of reducing false positive rate more rapidly than detection rate
• Example application – pedestrians walking
• sequences of 2,000 frames
• Each of the cascade classifiers trained on 2,250 positive and 2,250 negative examples
• each example – two $20 \times 15$ image windows from two consecutive image frames ($\delta t = 1$)

• positive examples – scaled bounding boxes of pedestrians
• negative examples – no pedestrians
• feature selection – 54,624 filters
• motion information was crucial for the achieved performance

• dynamic pedestrian detector clearly outperformed the static pedestrian detector
16.4 Detection of specific motion patterns

Figure 16.14: Example results of pedestrian detection using the dynamic pedestrian detector. Courtesy of P. Viola, Microsoft Live Labs and M. Jones, Mitsubishi Electric Research Labs, ©2003 IEEE [?].