Pixel Labeling: Stereo Vision

Lecture 18

See Material in
Reinhard Klette: Concise Computer Vision
Springer-Verlag, London, 2014

ccv.wordpress.fos.auckland.ac.nz

\(^1\)See last slide for copyright information.
Agenda

1 Pixel Interaction
2 Data Cost
3 Belief-Propagation Matching
4 Third-Eye Technique
Smoothness Cost Defines Pixel Interaction

Stereo matcher assigns disparity $f_p$ to pixel location $p \in \Omega$

$$E_{data}(p, f_p) = \text{dissimilarity cost (error) between}$$
neighbourhood around $p$ in base image $B$ and
neighbourhood around pixel in match image $M$ defined by disparity $f_p$

$$E_{smooth}(f_p, f_q) = \text{dissimilarity cost (error) between}$$
disparity $f_p$ at $p$ and
disparity $f_q$ at location $q$ adjacent to $p$

Stereo matcher aims at minimising the total error (or energy)

$$E(f) = \sum_{p \in \Omega} \left[ E_{data}(p, f_p) + \sum_{q \in A(p)} E_{smooth}(f_p, f_q) \right]$$

Consideration of adjacent pixel locations (assuming an iterative stereo matcher) defines growing regions of influence around each pixel
Markov, Bayes, Gibbs, and Pixel-interaction

The Russian mathematician A. A. Markov (1856 – 1922) studied stochastic processes where the interaction of multiple random variables can be modeled by an undirected graph. These models are today known as Markov random fields (MRFs).

If the underlying graph is directed and acyclic, then we have a Bayesian network, named after the English mathematician T. Bayes (1701 – 1761).

If we only consider strictly positive random variables then an MRF is called a Gibbs random field, named after the US-American scientist J. W. Gibbs (1839 – 1903).

Here: Error- (or energy-) minimisation by pixel-interaction on undirected pixel-adjacency graphs; labels assigned to pixels play the role of random variables; assigned labels and pixel-interaction specify an MRF model.
Agenda

1. Pixel Interaction

2. Data Cost

3. Belief-Propagation Matching

4. Third-Eye Technique
We Recall SSD and SAD

\[ p = (x, y) \text{ and } q = (x + d, y) \]

**SSD data cost measure**

\[
E_{SSD}(p, d) = \sum_{i=-l}^{l} \sum_{j=-k}^{k} [B(x + i, y + j) - M(x + d + i, y + j)]^2
\]

SSD for *sum of squared differences*

**SAD data cost measure**

\[
E_{SAD}(p, d) = \sum_{i=-l}^{l} \sum_{j=-k}^{k} |B(x + i, y + j) - M(x + d + i, y + j)|
\]

SAD for *sum of absolute differences*
Five Reasons Why Just SSD or SAD Will Not Work

1. *Invalidity of Intensity Constancy Assumption* (ICA). Intensity values at corresponding pixels, and in their neighborhoods, typically impacted by lighting variations, or just by image noise.

2. *Local reflectance differences*. Due to different viewing angles, $P$ and its neighborhood reflect light differently to cameras recording $B$ and $M$.

3. *Differences in cameras*. Different gains or offsets in cameras used result in high SAD or SSD errors.

4. *Perspective distortion*. 3D point $P = (X, Y, Z)$ is on a sloped surface; local neighborhood around $P$ on this surface is differently projected into images $B$ and $M$.

5. *No unique minimum*. There might be several pixel locations $q$ defining the same minimum.
Zero-Mean Version

Calculate mean $\bar{B}_x$ of a used window $W_{x}^{l,k}(B)$, and mean $\bar{M}_{x+d}$ of window $W_{x+d}^{l,k}(M)$, subtract $\bar{B}_x$ from all values in $W_{x}^{l,k}(B)$, and $\bar{M}_{x+d}$ from all values in $W_{x+d}^{l,k}(M)$, calculate this way the data-cost function in its zero-mean version.

Option for reducing impact of lighting artifacts (i.e. not depending on ICA)

Indicated by starting subscript of data-cost function with a $Z$

**Example:** $E_{ZSSD}$ or $E_{ZSAD}$ are zero-mean SSD or zero-mean SAD data-cost functions

$$E_{ZSSD}(x, d) = \sum_{i=-l}^{l} \sum_{j=-k}^{k} \left[ (B_{x+i,y+j} - \bar{B}_x) - (M_{x+i+d,y+j} - \bar{M}_{x+d}) \right]^2$$

$$E_{ZSAD}(x, d) = \sum_{i=-l}^{l} \sum_{j=-k}^{k} \left| [B_{x+i,y+j} - \bar{B}_x] - [M_{x+d+i,y+j} - \bar{M}_{i+d}] \right|$$
NCC Data Cost

Normalized cross correlation (NCC) useful for comparing two images

Already defined by zero-mean normalization, but we add $Z$ to the index for uniformity in notation:

$$E_{ZNCC}(x, d) = 1 - \frac{\sum_{i=-l}^{l} \sum_{j=-k}^{k} \left[ B_{x+i, y+j} - \overline{B}_x \right] \left[ M_{x+d+i, y+j} - \overline{M}_{x+d} \right]}{\sqrt{\sigma^2_{B, x} \cdot \sigma^2_{M, x+d}}}$$

where

$$\sigma^2_{B, x} = \sum_{i=-l}^{l} \sum_{j=-k}^{k} \left[ B_{x+i, y+j} - \overline{B}_x \right]^2$$

$$\sigma^2_{M, x+d} = \sum_{i=-l}^{l} \sum_{j=-k}^{k} \left[ M_{x+d+i, y+j} - \overline{M}_{x+d} \right]^2$$
Census Data-Cost Function

The zero-mean normalized census cost function

\[
E_{ZCEN}(x, d) = \sum_{i=-l}^{l} \sum_{j=-k}^{k} \rho(x + i, y + j, d)
\]

with

\[
\rho(u, v, d) = \begin{cases} 
0 & B_{uv} \perp B_x \text{ and } M_{u+d,v} \perp M_{x+d} \\
1 & \text{otherwise}
\end{cases}
\]

where \( \perp \) either < or >

By using \( B_x \) instead of \( \overline{B}_x \), and \( M_{x+d} \) instead of \( \overline{M}_{x+d} \), we have the census data-cost function \( E_{CEN} \) instead of \( E_{ZCEN} \)
Example for Census Data Cost

Windows $W_x(B)$ and $W_{x+d}(M)$

<table>
<thead>
<tr>
<th></th>
<th>2</th>
<th>1</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

|   | 5 | 5 | 9 |
|---|---|---|
| 1 | 2 | 4 |
| 5 | 4 | 6 |

Have $\overline{B}_x \approx 2.44$ and $\overline{M}_{x+d} \approx 6.11$

$i = j = -1$ results in $u = x - 1$ and $v = y - 1$

$B_{x-1,y-1} = 2 < 2.44$ and $M_{x-1+d,y-1} = 5 < 6.11$

Thus $\rho(x - 1, y - 1, d) = 0$

$i = j = +1$ results in $u = x + 1$ and $v = y + 1$

$B_{x+1,y+1} = 3 > 2.44$ but $M_{x+1+d,y+1} = 6 < 6.11$

Thus $\rho(x + 1, y + 1, d) = 1$

$i = j = -1$: values in the same relation with respect to the mean

$i = j = +1$: opposite relationships
Result for Example

For the given example: $E_{ZCEN} = 2$

Spatial distribution of $\rho$-values

\[
\begin{array}{ccc}
0 & 0 & 0 \\
1 & 0 & 0 \\
0 & 0 & 1 \\
\end{array}
\]

Vector $c_{x,d}$ lists these $\rho$-values in a left-to-right, top-to-bottom order:

\[ [0, 0, 0, 1, 0, 0, 0, 0, 1]^T \]
Hamming Distance

Let $b_x$ be the vector listing results $\text{sgn}(B_{x+i,y+j} - B_x)$ in a left-to-right, top-to-bottom order, where $\text{sgn}$ is the signum function.

Similarly, $m_{x+d}$ lists values $\text{sgn}(M_{x+i+d,y+j} - M_{x+d})$.

For the values in previous example

$$b_x = [-1, -1, +1, -1, -1, +1, -1, -1, +1]^T$$
$$m_{x+d} = [-1, -1, +1, +1, -1, +1, -1, -1, -1]$$
$$c_{x,d} = [0, 0, 0, 1, 0, 0, 0, 0, 0, 1]^T$$

Vector $c_{x,d}$ shows positions where $b_x$ and $m_{x+d}$ differ; the number of positions where two vectors differ is known as **Hamming distance**.
Efficient Calculation

**Observation** The zero-mean normalized census data cost \( E_{ZCEN}(x, d) \) equals the Hamming distance between vectors \( b_x \) and \( m_{x+d} \)

By replacing values “-1” by “0” in vectors \( b_x \) and \( m_{x+d} \), Hamming distance for resulting binary vectors can be calculated very time-efficiently
# Agenda

1. Pixel Interaction
2. Data Cost
3. Belief-Propagation Matching
4. Third-Eye Technique
BP as a General Optimization Framework

Introduced before in the context of image segmentation

Assign labels to all pixel locations based on given data and smoothness-cost terms, using a message-passing process

Belief-propagation matching (BPM) became very popular due to a paper (and provided sources) by P. F. Felzenszwalb and D. P. Huttenlocher, 2006

BPM solves the stereo-matching problem by pixel labeling, having \( \Omega \) (in the base image \( B \)) as the set of sites which will receive a label in the set \( L = \{0, 1, \ldots, d_{\text{max}}\} \)
Specific Data-Cost Terms in the Error Function

Data-cost functions such as ZCEN, ZSAD, etc. (avoiding the intensity constancy assumption) in the general pairwise MRF approach

\[ E(f) = \sum_{p \in \Omega} \left( E_{data}(p, f_p) + \sum_{(p,q) \in A} E_{smooth}(f_p - f_q) \right) \]

Smoothness-cost function is unary (a general term such as single-level or two-level Potts, linear, truncated square, etc.)

BPM: short for belief-propagation matching (for stereo vision)
BPM Example, Part 1

Assume two $5 \times 7$ images forming a stereo pair

Let $B = L$ and $d_{max} = 3$; search for corresponding pixel for $(x, y)$
BPM Example, Part 2

We have four message boards, all of size $5 \times 7$; we do not show $d_{\text{max}}$ columns left of $x$.

![Iterations with message passing](image)

Final message boards:

- $x$
  - $d=0$
  - $d=1$
  - $d=2$
  - $d=3$
Initialization of the Boards

Pixel \((x, y)\) in left image has potentially \(d_{\text{max}} + 1\) matching pixels \((x, y)\), \((x - 1, y)\), \((x - 2, y)\), and \((x - 3, y)\) in right image.

We have \(d_{\text{max}} + 1\) message boards.

Disparity between pixels \((x, y)\) and \((x, y)\) equals 0; thus, the cost for assigning disparity 0 to pixel \((x, y)\) is at position \((x, y)\) in Board 0.

Disparity between pixels \((x, y)\) and \((x - 1, y)\) equals 1; thus, the cost for assigning disparity 1 to pixel \((x, y)\) is at position \((x, y)\) in Board 1 – ETC.

Initially we insert data cost values into position \((x, y)\) for all the \(d_{\text{max}} + 1\) message boards.

Data cost value differences \(A = E_{\text{data}}((x, y), 0)\), analogously \(B\), \(C\), and \(D\), go into the four message boards at position \((x, y)\).
Start

Start at $t = 1$ and send messages between adjacent pixels

Pixel $p$ sends message vector of length $d_{\text{max}} + 1$ to any adjacent pixel, with message values for $d \in L$ in its $d_{\text{max}} + 1$ components

Let $m_{p \rightarrow q}^t$ be such a message vector, send from pixel at $p$ to adjacent pixel at $q$ in iteration $t$

For $d \in L$ we have message-update

$$m_{p \rightarrow q}^t(d) = \min_{h \in L} \left( E_{\text{data}}(p, h) + E_{\text{smooth}}(h - d) + \sum_{s \in A(p) \setminus q} m_{s \rightarrow p}^{t-1}(h) \right)$$

Accumulate at $q$ all messages from adjacent pixel locations $p$ (accumulated cost)

$$E_{\text{data}}(q, d) + \sum_{p \in A(q)} m_{p \rightarrow q}^t(d)$$

at pixel location $q$ for assigning disparity $d$ to $q$ at time $t$
BPM Example, Part 3

After a number of iterations, the iteration stops and defines new cost values $A'$, $B'$, $C'$, and $D'$ in the message boards at location $(x, y)$.

The minimum of those cost values defines the disparity (i.e., the label) which will be assigned to pixel at $(x, y)$.

For example, if $B' = \min\{A', B', C', D'\}$, then we have disparity 1 for the pixel at $(x, y)$ in the left image.
Pyramidal BPM

Use a regular pyramidal data structure for left and right image

Shortens distances between pixels for message passing

Use $k > 1$ layers in the two pyramids, with the first (the bottom) layer being the original image

Message boards are also transformed into a corresponding $k$-layer data structure

We initialize with data cost, in the first layer and also for the other $k - 1$ layers

Adjacency set of a pixel location in one of the layers contains now also pixel locations in adjacent layers, with connections defined by the regular generation of the pyramids

Now we follow exactly the BPM message-passing process
ICA Often Invalid for Real-World Data

Stereo pair, Sobel edge images, and BPM (with AD and Potts) results: left for original data, right for Sobel edge images
Asymmetry when Passing Step-Edges

“Strength” of message passing, from low-contrast areas in image $B$ to high-contrast areas, is less than the “strength” of message passing from a high-contrast area to a low-contrast area.

BPM is “fine” with generating consistent labels (disparities) in textureless regions, but may have difficulties when textures change.

In dependency of the chosen data and smoothness-cost functions, message passing can be “blocked” more or less by step-edges.

Often a positive effect of preserving depth discontinuities at intensity discontinuities.
Agenda

1. Pixel Interaction
2. Data Cost
3. Belief-Propagation Matching
4. Third-Eye Technique
Evaluation of Stereo Matchers

One Option: **Third-Eye Technique - Basic Outline**

1. Record stereo data with two cameras and calculate disparities
2. Have also a third calibrated camera looking into the same space as the other two cameras
3. Use the calculated disparities for mapping the recorded image of the (say) left camera into the image plane of the third camera, thus creating a virtual image
4. Compare the virtual image with the image recorded by the third camera

If the virtual and the third image “basically coincide” then the stereo matcher provides “useful” disparities
Third and Virtual Image

*Left:* Third image  
*Right:* Virtual image, to be compared with the third image
Virtual Images for Different Matchers

Test of BPM by using the same stereo matcher on three different input data: original data (left), residuals with respect to smoothing (right), and Sobel edge maps (not shown)
Normalised Cross Correlation Third vs. Virtual Image
Comments for Shown Example

The example illustrates an application of the third-eye technique for comparing BPM on a given video sequence of 120 frames.

The scale for NCC is in percent.

The pyramidal BPM algorithm used the simple AD data cost function, which is based on the ICA, and this is an incorrect assumption for real-world recording as for the illustrated sequence.

Two methods for data preprocessing are used, Sobel edge maps or residuals with respect to smoothing (another option: use ZCEN data cost on original data).

NCC only calculated in a masked set $\Omega_t$ which contains only pixels being in a distance of at most 10 from the closest Canny edge pixel.
NCC Diagrams Identify Research Issues

The NCC diagram indicates a significant drop in stereo-matching performance about at Frame 60.

This is one of the important opportunities provided by the third-eye technique: identify those situations where recorded video sequences cannot be processed properly by a given stereo matcher, and start your research into the question how to resolve the issue for the given situation.

Actually: different stereo matchers fail typically at the same place in a recorded sequence (to different extends), thus identifying a *generic stereo-matching issue* at that place.
Copyright Information

This slide show was prepared by Reinhard Klette with kind permission from Springer Science+Business Media B.V.

The slide show can be used freely for presentations. However, all the material is copyrighted.


In case of citation: just cite the book, that’s fine.