Building Extraction from Digital Elevation Models

U. Weidner
e-mail: weidner@ipb.uni-bonn.de

BEX Manual
Version:
February 1995

Printed:

May 11, 1995

Institut für Photogrammetrie
Universität Bonn
Nußallee 15
D 53115 Bonn
Contents

A User’s Guide 3
A.1 How to use BEX ................................. 3
A.2 Input Data .................................. 3
A.3 Output Files ................................. 6
A.4 Trouble Shooting .......................... 11

B Programmer’s Guide 14
B.1 Flow Chart .................................. 14
B.2 Header File ................................. 18

List of Figures

B-1 Flow chart .................................. 17

List of Tables

A-1 General input parameters .................. 3
A-2 File formats ................................. 4
A-3 Output files: parametric models ........ 7
A-4 Output files: prismatic models .......... 8
A-5 Output files: additional data .......... 8
A-6 Error codes I ............................. 12
A-7 Error codes II ............................ 13
A-8 Error codes III ............................ 13
<table>
<thead>
<tr>
<th>Input parameter</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>input filename</td>
<td>cf. Remark A.1</td>
</tr>
<tr>
<td>pixel size in x-direction</td>
<td>cf. Remark A.2</td>
</tr>
<tr>
<td>pixel size in y-direction</td>
<td>cf. Remark A.3</td>
</tr>
<tr>
<td>resolution in height</td>
<td>cf. Remark A.4</td>
</tr>
<tr>
<td>first label</td>
<td>cf. Remark A.5</td>
</tr>
<tr>
<td>output data format</td>
<td>cf. Remark A.6</td>
</tr>
<tr>
<td>output filename</td>
<td>cf. Remark A.7</td>
</tr>
<tr>
<td>parametric models</td>
<td>cf. Remark A.8</td>
</tr>
<tr>
<td>prismatic models</td>
<td>cf. Remark A.10</td>
</tr>
<tr>
<td>write grid models</td>
<td>cf. Remark A.11</td>
</tr>
<tr>
<td>write data for each model</td>
<td>cf. Remark A.12</td>
</tr>
<tr>
<td>write additional data</td>
<td>cf. Remark A.13</td>
</tr>
<tr>
<td>show results</td>
<td></td>
</tr>
</tbody>
</table>

Tab. A-1: General input parameters

A  User’s Guide

A.1  How to use BEX

BEX needs a control parameter file as given in the next section. Therefore, BEX can be used in two different ways:

```
bex
```

invokes the creation of a control parameter file `bex.ini`, which has to be edited by the user. If this file already exists, the program asks the user to edit the file.

```
bex  control parameter filename
```

runs the program with the control parameters in the named file.

The following sections describe the input data (Section A.2), the output (Section A.3), and error messages of the program (Section A.4).

A.2  Input Data

Table A-1 shows further input parameters which have not been discussed. An example of a control parameter file is given in the next paragraph.
### Tab. A-2: File formats

**Remark A.1** input filename specifies the name of the DEM-file. This file has to be in a format of those given in Table A-2. **MATCH-T ASCII-files** have to be converted using the programm **inpho**. For this, all grid points have to be defined in the **MACTH-T ASCII-file**.

**Remark A.2** Different sizes are allowed.

**Remark A.3** Up to now this parameter is not needed.

**Remark A.4** The user can fix the first label for the segments.

**Remark A.5** Different output formats for the files *.*data* and *.*build* are possible (cf. Table A-2) It is recommended to use the format **IPB-FLOAT** (51).

**Remark A.6** output filename specifies the file name for the output files. Extensions are set automatically. See Table A-3, Table A-4 and Table A-5.

**Remark A.7** Option parametric models: Inputs are

- 0 no parametric models
- 1 compute parametric models

For output files see Table A-3.

**Remark A.8** Option prismatic models: Inputs are

- 0 no prismatic models
- 1 compute prismatic models
If only prismatic models are selected, the program searches for existing files

- *.bound_box.dat
- *.valid_label.dat
- *.int_seg
- *.int_seg_lines

If these files exist in the current directory, only the ground plan extraction is performed without new detection of buildings. For output files see Table A-4.

Remark A.9 Pure detection of buildings is possible using 0 for the control parameters parametric models and prismatic models.

Remark A.10 Option write grid models: Inputs are

\[
\begin{align*}
0 & \text{ do not write grid models} \\
1 & \text{ write grid models}
\end{align*}
\]

This option is only valid for parametric models.

Remark A.11 Option write data for each model: Inputs are

\[
\begin{align*}
0 & \text{ do not write data for each model} \\
1 & \text{ write data for each model}
\end{align*}
\]

This option is valid for both groups of models.

Remark A.12 Option write additional data: Inputs are

\[
\begin{align*}
0 & \text{ do not write additional data} \\
1 & \text{ write additional data}
\end{align*}
\]

For output files see Table A-5. Some of the files are only written, if the corresponding option for the models is set.

Remark A.13 Option show results: Inputs are

\[
\begin{align*}
0 & \text{ do not show (some of) the results} \\
1 & \text{ show (some of) the results}
\end{align*}
\]

The results are shown using visualization tools of IPB. During visualization the program is stopped.
Example of Control Parameter File

# control parameter file BEX

flat_05.dhm # data
  0.5   # pixel_size in x-direction [m]
  0.5   # pixel_size in y-direction [m]
78909.0 # x-coordinate (lower left corner) [m]
44981.0 # y-coordinate (lower left corner) [m]
50.0   # expected size of buildings [m^2]
500.0  # area min/max filter
  4.0   # height for initial segmentation
  3.0   # minimal slope [degree]
  0.1   # resolution in height
  0.5   # threshold for polygon (triangle height) [m]
100    # first label
  2     # additional margin for bounding box [pixel]
  51    # output data format for grid models of buildings
bextestf # name (without extension)
  1     # parametric models
  1     # prismatic models
  1     # write grid models
  1     # write data for each model
  1     # write additional data
  0     # show results

A.3 Output Files

Lists of Output Files The output files depend on the control parameters parametric models, prismatic models, write grid models, write data for each model, and write additional data.

- The output files for parametric models are listed in Table A-3.
- The output files for prismatic models are listed in Table A-4.
- The option write grid models is only valid if parametric models are selected. The files name.build.label contain the grid models for each label.
- The option write data for each model is valid for both, either parametric models or prismatic models. The files name.data.label contain the original data within the bounding box for each label.
<table>
<thead>
<tr>
<th>File name</th>
<th>Contents</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>name.bound_box_dat</td>
<td>file with bounding box information</td>
<td>A</td>
</tr>
<tr>
<td>name.build.label</td>
<td>file of the grid model of each label</td>
<td>*</td>
</tr>
<tr>
<td>name.build_lines</td>
<td>file for the visualization of the extracted models’ outlines</td>
<td>IPBU</td>
</tr>
<tr>
<td>name.dem_build</td>
<td>file for the visualization of the result: data and grid models of the extracted buildings</td>
<td>*</td>
</tr>
<tr>
<td>name.hase</td>
<td>file for HASE-parameters</td>
<td>A</td>
</tr>
<tr>
<td>name.intseg</td>
<td>file for the visualization of the refined segments</td>
<td>**</td>
</tr>
<tr>
<td>name.intseg_bin</td>
<td>file for the visualization of the refined segments</td>
<td>IPBU</td>
</tr>
<tr>
<td>name.intseg_lines</td>
<td>file for the visualization of the refined segments’ outlines</td>
<td>IPBU</td>
</tr>
<tr>
<td>name.labels</td>
<td>file for the visualization of the labels</td>
<td>**</td>
</tr>
<tr>
<td>name.prt</td>
<td>file of output parameters</td>
<td>A</td>
</tr>
<tr>
<td>name.segments</td>
<td>file of the initial segmentation</td>
<td>**</td>
</tr>
<tr>
<td>name.sel_labels</td>
<td>file for the visualization of the valid labels</td>
<td>**</td>
</tr>
<tr>
<td>name.valid_label.dat</td>
<td>file with code number of labels</td>
<td>A</td>
</tr>
</tbody>
</table>

**Tab. A-3: Output files: parametric models**

- The option *write additional data* is also valid for both groups of models. The output files are listed in Table A-5. The actual files also depend on the options *parametric models* and *prismatic models*.

- The formats marked in the tables are

  A  ASCII file
  P  POLY file
  IPBU  IPB-format using *unsigned character*
  IPBF  IPB-format using *float*
  *  format is given by the user
  **  format is determined by the number of labels, either IPBU or IPBF
  *** format of the data file
### Tab. A-4: Output files: prismatic models

<table>
<thead>
<tr>
<th>File name</th>
<th>Contents</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>name.bound_box</td>
<td>file with bounding box information</td>
<td></td>
</tr>
<tr>
<td>name.first_poly</td>
<td>file of polygons (vectorization of outlines)</td>
<td>P</td>
</tr>
<tr>
<td>name.first_poly_label</td>
<td>file of polygon (vectorization of outline) of each label</td>
<td>P</td>
</tr>
<tr>
<td>name.int_seg</td>
<td>file for the visualization of the refined segments</td>
<td>**</td>
</tr>
<tr>
<td>name.int_seg_bin</td>
<td>file for the visualization of the refined segments</td>
<td>IPBU</td>
</tr>
<tr>
<td>name.int_seg_lines</td>
<td>file for the visualization of the refined segments' outlines</td>
<td>IPBU</td>
</tr>
<tr>
<td>name.int_seg_lines_label</td>
<td>file for the visualization of the refined segments' outlines of each label</td>
<td>IPBU</td>
</tr>
<tr>
<td>name.labels</td>
<td>file for the visualization of the labels</td>
<td>**</td>
</tr>
<tr>
<td>name.prt</td>
<td>file of output parameters</td>
<td>A</td>
</tr>
<tr>
<td>name.ref_poly</td>
<td>file of refined polygons</td>
<td>P</td>
</tr>
<tr>
<td>name.ref_poly_label</td>
<td>file of refined polygons of each label</td>
<td>P</td>
</tr>
<tr>
<td>name.segments</td>
<td>file of the initial segmentation</td>
<td>**</td>
</tr>
<tr>
<td>name.sel_labels</td>
<td>file for the visualization of the valid labels</td>
<td>**</td>
</tr>
<tr>
<td>name.valid_label</td>
<td>file with code number of labels</td>
<td>A</td>
</tr>
</tbody>
</table>

### Tab. A-5: Output files: additional data

<table>
<thead>
<tr>
<th>File name</th>
<th>Contents</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>name.bound_box</td>
<td>file for the visualization of bounding boxes</td>
<td>**</td>
</tr>
<tr>
<td>name.build</td>
<td>file of all grid models</td>
<td>IPBF</td>
</tr>
<tr>
<td>name.build_lines</td>
<td>file for the visualization of the extracted models' outlines in the ((x,y))-plane</td>
<td>IPBU</td>
</tr>
<tr>
<td>name.dem_build</td>
<td>file for the visualization of the result: data and grid models of the extracted buildings</td>
<td>IPBF</td>
</tr>
<tr>
<td>name.difference</td>
<td>difference between data and filtered data</td>
<td>***</td>
</tr>
<tr>
<td>name.min</td>
<td>minimum filtered data</td>
<td>***</td>
</tr>
<tr>
<td>name.minmax</td>
<td>minimum-maximum filtered data</td>
<td>***</td>
</tr>
<tr>
<td>name.num_poly_label</td>
<td>file of polygons (number)</td>
<td>IPBU</td>
</tr>
<tr>
<td>name.poly</td>
<td>file of polygons (straight lines)</td>
<td>P</td>
</tr>
<tr>
<td>name.poly_label</td>
<td>file of polygon (straight lines) of each label</td>
<td>P</td>
</tr>
<tr>
<td>name.pose</td>
<td>file for the visualization of the extracted points of gravity and main axes</td>
<td>IPBU</td>
</tr>
<tr>
<td>name.sel_data</td>
<td>file containing only the data of the valid labels</td>
<td>***</td>
</tr>
<tr>
<td>name.size_of_label</td>
<td>file for the visualization of the initial segments' sizes</td>
<td>IPBF</td>
</tr>
</tbody>
</table>
Example of Output File

BEX report file

data file : flat05.dhm
number of rows : 481
cols : 481

input variables

pixel size in x direction [m] : 0.5000
pixel size in y direction [m] : 0.5000
x-coordinate (lower left corner) [m] : 78909.0000
y-coordinate (lower left corner) [m] : 44981.0000
minimal size of buildings [m^2] : 50.0000
area min/max filter [m^2] : 500.0000
height for initial segmentation [m] : 4.0000
minimal slope [degree] : 3.0000
resolution in height [m] : 0.1000
threshold for refined polygon [m] : 0.5000
first label : 100
margin for bounding box [pixel] : 2

output variables

pixel area [m^2] : 0.2500
width for min/max-filter : 45
last label : 139
number of labels : 40
number of valid labels : 20

output for labels

...
col : 187
number of cols : 51
height min of bobo : 100.6383
max : 111.2185
min of label : 105.9378
max : 111.2185
of interest : 105.9284
mean of segment : 108.4103
mean of box-segment : 101.7229
size of interest segment [pixel] : 1167
[m^2] : 291.7500
outline length of segment [pixel] : 172
number of polygons : 1
number of polygon points : 17

internal coordinate system
point of gravity r [pixel] : 347.7673
c [pixel] : 213.5216
orientation phi [degree] : 36.3166

object coordinate system
reference point x [m] : 79008.2344
y [m] : 45057.8516
z [m] : 101.7229
orientation [degree] : 143.6834

building length [m] : 25.4000
width [m] : 13.9000
height 1 [m] : 4.2149
height 2 [m] : 5.2807
size [m^2] : 353.0600
volume [m^3] : 2420.3057

.
.
.

label no. 139
non valid label : bounding box
A.4 Trouble Shooting

Some of the error messages are given by code numbers, other are printed on the screen:

existing files
  bex searches for existing output files of the specified output filename. If these files exist, the user has to move or delete these files. bex cannot search for those files (A, P) which correspond to a label. Nevertheless, the existence of such a file causes an exit of the program.

memory allocation
  bex allocates a lot of memory for the computation, which is also freed again, if the information is not needed any longer. If an allocation of memory fails, the program will give a message on the screen, which indicates, which allocation failed (vector, matrix) and the calling subroutine, and the program will exit.

control parameterfile While reading the control parameterfile, errors may occur. Some of them are indicated by a short message and a code number. These code numbers are given in Table A-6, Table A-7 and Table A-8.
<table>
<thead>
<tr>
<th>Code</th>
<th>Error reading</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>pixel size in x-direction</td>
</tr>
<tr>
<td>12</td>
<td>pixel size in y-direction</td>
</tr>
<tr>
<td>13</td>
<td>x-coordinate (lower left)</td>
</tr>
<tr>
<td>14</td>
<td>y-coordinate (lower left)</td>
</tr>
<tr>
<td>21</td>
<td>expected minimum size of building</td>
</tr>
<tr>
<td>22</td>
<td>area for min/max filter</td>
</tr>
<tr>
<td>23</td>
<td>height for initial segmentation</td>
</tr>
<tr>
<td>24</td>
<td>minimal slope</td>
</tr>
<tr>
<td>25</td>
<td>resolution in height</td>
</tr>
<tr>
<td>29</td>
<td>threshold for refined polyon</td>
</tr>
<tr>
<td>31</td>
<td>first label</td>
</tr>
<tr>
<td>32</td>
<td>additional margin for bounding box</td>
</tr>
<tr>
<td>33</td>
<td>output data format</td>
</tr>
<tr>
<td>34</td>
<td>option parametric models</td>
</tr>
<tr>
<td>35</td>
<td>option prismatic models</td>
</tr>
<tr>
<td>36</td>
<td>option write grid models</td>
</tr>
<tr>
<td>37</td>
<td>option write data for each model</td>
</tr>
<tr>
<td>38</td>
<td>option write additional data</td>
</tr>
<tr>
<td>39</td>
<td>option show results</td>
</tr>
<tr>
<td>80</td>
<td>data filename</td>
</tr>
<tr>
<td>81</td>
<td>data file</td>
</tr>
<tr>
<td>82</td>
<td>output filename</td>
</tr>
</tbody>
</table>

Tab. A-6: Error codes I
<table>
<thead>
<tr>
<th>Code</th>
<th>Error during plausibility check . . .</th>
<th>Allowed values . . .</th>
</tr>
</thead>
<tbody>
<tr>
<td>111</td>
<td>pixel size in x-direction</td>
<td>&gt; 0</td>
</tr>
<tr>
<td>112</td>
<td>pixel size in y-direction</td>
<td>&gt; 0</td>
</tr>
<tr>
<td>113</td>
<td>x-coordinate (lower left)</td>
<td></td>
</tr>
<tr>
<td>114</td>
<td>y-coordinate (lower left)</td>
<td></td>
</tr>
<tr>
<td>121</td>
<td>expected minimum size of building</td>
<td>&gt; 0</td>
</tr>
<tr>
<td>122</td>
<td>are for min/max filter</td>
<td>&gt; size of building</td>
</tr>
<tr>
<td>123</td>
<td>height for initial segmentation</td>
<td>&gt; 0</td>
</tr>
<tr>
<td>124</td>
<td>minimal slope</td>
<td>&gt; 0</td>
</tr>
<tr>
<td>125</td>
<td>resolution in height</td>
<td>≥ 0</td>
</tr>
<tr>
<td>129</td>
<td>threshold for refined polyon</td>
<td>&gt; 0</td>
</tr>
<tr>
<td>131</td>
<td>first label</td>
<td>&gt; 0</td>
</tr>
<tr>
<td>132</td>
<td>additional margin for bounding box</td>
<td>≥ 0</td>
</tr>
<tr>
<td>133</td>
<td>output data format not allowed</td>
<td>cf. Table A-2</td>
</tr>
<tr>
<td>134</td>
<td>option parametric models not allowed</td>
<td>0, 1</td>
</tr>
<tr>
<td>135</td>
<td>option prismatic models not allowed</td>
<td>0, 1</td>
</tr>
<tr>
<td>136</td>
<td>option write grid models not allowed</td>
<td>0, 1</td>
</tr>
<tr>
<td>137</td>
<td>option write data for each model not allowed</td>
<td>0, 1</td>
</tr>
<tr>
<td>138</td>
<td>option write additional data not allowed</td>
<td>0, 1</td>
</tr>
<tr>
<td>139</td>
<td>option show results not allowed</td>
<td>0, 1</td>
</tr>
</tbody>
</table>

**Tab. A-7: Error codes II**

<table>
<thead>
<tr>
<th>Code</th>
<th>Error during file check . . .</th>
</tr>
</thead>
<tbody>
<tr>
<td>801</td>
<td>output file *.prt exists</td>
</tr>
<tr>
<td>802</td>
<td>output file *.hase exists</td>
</tr>
<tr>
<td>803</td>
<td>output file *.first_poly exists</td>
</tr>
<tr>
<td>804</td>
<td>output file *.ref_poly exists</td>
</tr>
<tr>
<td>805</td>
<td>output file *.poly exists</td>
</tr>
<tr>
<td>806</td>
<td>output file *.bound_box_dat exists</td>
</tr>
<tr>
<td>807</td>
<td>output file *.valid_label_dat exists</td>
</tr>
<tr>
<td>888</td>
<td>EOF of control parameter file not found</td>
</tr>
</tbody>
</table>

**Tab. A-8: Error codes III**
B  Programmer’s Guide

B.1  Flow Chart

bex

→ bex_in
    task: read control parameter file
    alloc: bex.d
    free: -

if shortcut

→ bex_seg
    task: compute initial segmentation
    label segments
    compute segment size
    compute bounding box
    by: minimum filtering
    maximum filtering
    computing difference
    thresholding
    computing connected components
    alloc: bex.dmin
    bex.dminmax
    bex.ddif
    bex.seg
    bex.labels
    free: bex.dmin
    bex.dminmax
    bex.ddif
    bex.seg

→ bex_sob
    task: compute size of labels
    alloc: -
    free: -

→ bex_bobo
    task: compute bounding boxes
    alloc: bex.box_r
    bex.box_c
    bex.box_dr
    bex.box_dc
    free: -
\textbf{\textit{bex\_seg}}

\texttt{task: select valid labels}
\texttt{alloc: bex.valid\_label}
\texttt{bex.rlabels}
\texttt{free: -}

\textbf{\textit{bex\_int}}

\texttt{task: compute refined segmentation}
\texttt{alloc: bex.max\_height}
\texttt{bex.min\_height}
\texttt{bex.max\_height\_wol}
\texttt{bex.min\_height\_wol}
\texttt{bex.mean\_height\_wol}
\texttt{bex.mean\_height\_wol}
\texttt{bex.max\_height\_wol}
\texttt{bex.height\_pf\_int}
\texttt{bex.size\_of\_intseg}
\texttt{bex.intseg}
\texttt{bex.intseg\_lines}
\texttt{bex.dint}
\texttt{free: bex.labels}
\texttt{bex.rlabels}

\textbf{if parametric models}

\textbf{\textit{bex\_para}}

\texttt{task: compute parameters for parametric models}
\texttt{alloc: bex.pos.x}
\texttt{bex.pos.y}
\texttt{bex.phi}
\texttt{bex.length}
\texttt{bex.width}
\texttt{bex.ridges\_height}
\texttt{bex.eaves\_height}
\texttt{bex.base\_height}
\texttt{bex.pose}
\texttt{free: bex.pose}
\texttt{bex.dint}

\textbf{if parametric models}

\textbf{\textit{bex\_hou}}

\texttt{task: compute grid models}
\texttt{alloc: -}
\texttt{free: -}

\textbf{if write data}

\textbf{\textit{bex\_data}}

\texttt{task: write data within bounding box for each model}
\texttt{alloc: -}
\texttt{free: -}
if parametric models

    → bex_hase
        task: compute parameters for program HASE
        alloc: bex.x_ref
               bex.y_ref
               bex.z_ref
               bex.ridges.eaves
               bex.eaves_base
               bex.ori_ref
        free: -

else

    read files
        files: *.bound_box.lat, *.valid_label.lat, *.int_seg, *.int_seg_lines
        alloc: bex.box.r
               bex.box.c
               bex.box.dr
               bex.box.dc
               bex.valid_label
               bex.intseg
               bex.intseg_lines

endif shortcut

    → bex_poly
        task: compute polygons
        alloc: bex.outline.intseg
               bex.ref.poly nop
               bex.ref_poly_num
        free: -

    → bex_out
        task: write output file
        alloc: -
        free: -

free: bex.intseg
      bex.intseg_lines
Fig. B-1: Flow chart
B.2 Header File

#define BEXSTRUCT static BEXTYP
typedef
struct
{
  /* matrices */
  /* input: */
  MATTYP d;            /* data */
  /* output: */
  MATTYP dmin;         /* MIN(data) */
  MATTYP dminmax;      /* MAX(MIN(data)) */
  MATTYP ddiff;        /* Difference data - MIN(MAX(data)) */
  MATTYP seg;          /* initial segments (is) */
  MATTYP labels;       /* labels of is */
  MATTYP rseg;         /* valid segments (vs) */
  MATTYP rlabels;      /* labels of vs */
  MATTYP intseg;       /* refined segmentation (rs) */
  MATTYP intseg_lines; /* outlines of rs */
  MATTYP dint;         /* reduced data */
  MATTYP pose;         /* position */
  MATTYP gb;           /* gradients (abs) */
  MATTYP gw;           /* gradients (ori) */
  MATTYP gvw;          /* gradients (var) */
  MATTYP hk1;          /* principal curvature 1 */
  MATTYP hk2;          /* principal curvature 2 */
  MATTYP hkw;          /* principal direction direction */
}
/* vectors */
/* output: */

VECTYP box_r;        /* r position of bounding box [pixel] */
VECTYP box_c;        /* c position of bounding box [pixel] */
VECTYP box_dr;       /* #rows of bounding box [pixel] */
VECTYP box_dc;       /* #cols of bounding box [pixel] */
VECTYP size_of_label; /* size of label (is) [pixel] */
VECTYP size_of_intseg; /* size of label (rs) [pixel] */
VECTYP max_height;   /* max height in bounding box */
VECTYP min_height;   /* min height in bounding box */
VECTYP mean_height_wol; /* mean height of background in bounding box */
VECTYP height_of_int; /* thresholds rs */
VECTYP max_height_1; /* max heights of labels */
VECTYP min_height_1; /* min heights of labels */
VECTYP mean_height_1; /* mean height of labels */
VECTYP ridges_height; /* height of ridges */
VECTYP eaves_height; /* height of eaves */
VECTYP base_height; /* height of base */
VECTYP length; /* length */
VECTYP width; /* width */
VECTYP eaves_base; /* height 1 -> HASE */
VECTYP ridges_eaves; /* height 2 -> HASE */
VECTYP x_ref; /* x-coord. reference point */
VECTYP y_ref; /* y-coord. reference point */
VECTYP z_ref; /* z-coord. reference point */
VECTYP ori_ref; /* orientation in reference system */
VECTYP area_of_label; /* Flaeche als robusste Mittelbildung */
VECTYP pos_r; /* r-coord. of point of gravity */
VECTYP pos_c; /* c-coord. of point of gravity */
VECTYP phi; /* orientation in rc-system */
VECTYP eigen1, eigen2; /* eigenvalues */
VECTYP schiefe; /* unsymmetry */

IVECTYP valid_label; /* code number */
IVECTYP outline_intseg; /* number of outline points */
IVECTYP ref_poly_num; /* number of polygons */
IVECTYP ref_poly_nop; /* number of polygon points */

/* variables */
/* input: */

float size_of_build; /* expected minimal size of building (m*m) */
float xo; /* lower left coord. in reference system */
float yo; /* lower left coord. in reference system */
float xsize_of_pixel; /* pixel size in x (m) */
float ysize_of_pixel; /* pixel size in y (m) */
float area_of_int; /* area MIN/MAX-filtering */

float segh; /* height for initial segmentation */
float slope; /* min slope */
float ridge_eave_h; /* difference ridge - eave */
float height_res; /* resolution of height */
float ref_poly_thresh; /* threshold for refined polygon */

int boxrand; /* additional margin for bounding box */
int first_label; /* first labels */
int param_flg; /* flag parametric models */
int poly_flg; /* flag prismatic models */
int out_flg; /* flag write additional data */
int build_flg; /* flag write grid models */
int data_flg; /* flag write data */
int steppoly; /* flag shortcut */

/* output: */

float size_of_pixel; /* size of pixel (m*m) */

int last_label; /* last label */
int number_of_labels; /* number of labels (is) */
int number_of_rlabels; /* number of labels (vs) */
int xminfilb;  /* window size MIN/Max */
int form;    /* format of input data file */
int form_md; /* format for output files *.build *.data */

/* filenames */
char *dfile;    /* data */
char *stfile;   /* control parameter */
char *prtfile;  /* output */
char *bexfile;  /* project */
char *file;     /* (additional) */

/* else */
char *kommentar; /* data comment */
FILE *fp;      /* output */
int iniflag;

int scr_flg;
char *scr_ipbu;
char *scr_ipbf;
char *scr_plot;
char *scr_cmd;
}
BEXTYP;

void bex_in();
void bex_out();
void bex_seg();
void bex_rseg();
void bex_int();
void bex_bobo();
void bex_sob();
void bex_para();
void bex_hou();
void bex_data();
void bex_has();
void bex_poly();
void bex_scr();