```
1
    classdef class videoinput < handle</pre>
        %CLASS WINVIDEO Communication with a camera via videoinput
 2
 3
        %
4
        properties (SetAccess = immutable)
            5
            type = '';
                             % videoinput type
 6
            device_name = ''; % like in imaqhwinfo(type,
 7
            id).DeviceName
            format = '';
                             % format for videoinput function
8
9
            color_space = ''; % ReturnedColorSpace
            id = false;
                             % ID-Number
10
11
        end
12
13
        properties (Access = private)
14
            prev timer = false;% timer for async snapshot preview
15
16
        end
17
18
        methods
19
            % constructor
            function obj = class videoinput(name, type, format,
20
            color_space, device_name)
21
                obj.name = name;
22
                obj.type = type;
23
                obj.format = format;
                obj.color space = color space;
24
25
                obj.device name = device name;
26
27
                % find camera id
28
                cam id = -1;
29
                info = imaghwinfo(obj.type);
30
                for i = info.DeviceIDs
31
                    dev_info = imaqhwinfo(obj.type, i{1});
                    if strcmp(dev info.DeviceName, device name)
32
                       cam id = i\{1\};
33
34
                    end
35
                end
36
                % return if device name is not found
37
38
                if cam id == -1
                    error('Can''t find camera %s by DeviceName
39
                    "%s"', name, device name);
40
                end
41
                % initialize camera
42
43
                obj.id = cam id;
                fprintf('Found videoinput camera %s at ID %d.\n',
44
                name, cam id);
45
            end
```

```
46
47
             % connect to the camera
48
             function success = connect(obj)
                 success = false;
49
50
                 if obj.handle ~= false
51
                      success = true;
52
                      return;
53
                 end
54
55
                 if obj.id == false
                     warning('Can''t connect to %s, don''t know
56
                      ID.', obj.name);
57
                      return;
58
                 end
59
60
                 try
                      obj.handle = videoinput(obj.type, obj.id,
61
                      obj.format);
62
                      obj.handle.ReturnedColorSpace = obj.color space;
63
                      fprintf('Connected to %s\n', obj.name);
64
                      success = true;
65
                 catch e
                      warning('Connecting to %s failed:
66
                      videoinput(''%s'', %d, ''%s''). %s', obj.type,
                      obj.name, obj.id, obj.format, getReport(e));
67
                 end
68
             end
69
70
             % close camera connection
71
             function close(obi)
72
                 if obj.handle ~= false
73
                      delete(obj.handle);
74
                      obj.handle = false;
75
                 end
76
                 fprintf('Closed connection to %s\n', obj.name);
77
             end
78
             % Call config function with handle if connected
79
             function [success, varargout] = config(obj,
80
             config function)
                 success = false;
81
82
                 varargout = cell(1, nargout - 1);
83
                 % warn and return if connecting failes
84
85
                 if obj.connect() == false
                      warning('Config error: Can''t connect to camera
86
                     %s.', obj.name);
87
                      return;
88
                 end
```

```
89
 90
                  try
                       if isa(config_function, 'function_handle')
 91
                           [obj.handle, varargout{1:nargout - 1}] =
 92
                           config function(obj.handle);
                       elseif iscell(config_function) &&
 93
                       isa(config_function{1}, 'function_handle')
                           [obj.handle, varargout{1:nargout - 1}] =
 94
                           config function{1}(obj.handle,
                           config function{2:end});
 95
                       else
                           error('videoinput config callback is not
 96
                           callable');
 97
                       end
 98
99
                       success = true;
100
                  catch e
                       warning('Exception while config %s: %s',
101
                       obj.name, getReport(e));
102
                  end
103
              end
104
              % preview live image
105
              function success = preview(obj, adjust function, axes,
106
              rotate)
107
                  success = false;
108
                  % warn and return if connecting failes
109
                  if obj.connect() == false
110
                       warning('Preview error: Can''t connect to
111
                       camera %s.', obj.name);
112
                       return;
113
                  end
114
115
                  try
                       size = obj.handle.ROIPosition;
116
                       bands = obj.handle.NumberOfBands;
117
118
                       if nargin > 3 && rotate == true
119
120
                           size([3, 4]) = size([4, 3]);
121
                       end
122
123
                       if nargin < 3</pre>
                           % preview as figure
124
                           hImage = image(zeros(size(4), size(3),
125
                           bands));
126
                       else
                           % preview on axes in GUI
127
                           hImage = image(zeros(size(4), size(3),
128
```

```
bands), 'Parent', axes);
129
                           axes.DataAspectRatio = [1, 1, 1];
130
                       end
131
                       if nargin > 1 && isa(adjust function,
132
                       'function handle')
133
                           setappdata(hImage,
                           'UpdatePreviewWindowFcn', adjust function);
134
                       end
135
                       preview(obj.handle, hImage);
136
137
138
                       success = true;
139
                   catch e
                       warning('Exception while preview %s: %s',
140
                       obj.name, getReport(e));
141
                   end
142
              end
143
144
              % stop preview live image
              function success = stoppreview(obj)
145
                   success = false:
146
                   if obj.handle == false
147
148
                       return;
149
                   end
150
                   try
151
152
                       if obj.prev timer ~= false
                           stop(obj.prev_timer);
153
                           delete(obj.prev_timer);
154
                           obj.prev_timer = false;
155
156
                       end
157
158
                       closepreview(obj.handle);
159
                       success = true;
160
                   catch e
                       warning('Exception while preview %s: %s',
161
                       obj.name, getReport(e));
162
                   end
163
              end
164
              function [success, img] = snapshot(obj,
165
              adjust function, axes)
                   success = false;
166
167
168
                   try
169
                       if nargin < 2</pre>
                           adjust function = false;
170
171
                       end
```

```
172
173
                       img = false;
174
175
                       obj.stoppreview();
176
                       % warn and return if connecting failes
177
                       if obj.connect() == false
178
                           warning('Snapshot error: Can''t connect to
179
                           camera %s.', obj.name);
180
                           return;
181
                       end
182
                       img = getsnapshot(obj.handle);
183
184
                       if nargin > 2
185
                           if isa(adjust_function, 'function_handle')
186
                               img = adjust_function(img);
187
188
                           end
189
                           imshow(img, 'Parent', axes);
190
                       else
191
                           imshow(img);
192
193
                       end
194
195
                       success = true;
                  catch e
196
197
                       warning('Exception while snapshot %s: %s',
                       obj.name, getReport(e));
198
                  end
199
              end
200
              function delete(obj)
201
                  obj.close();
202
203
              end
          end
204
205
      end
```

206