Pyforms GUI Documentation

Release 4.0

Ricardo Jorge Vieira Ribeiro

Mar 06, 2019

PyForms

PYFORMS

Pyforms is a Python 3 framework to develop applications capable of executing in 3 diferent environments, Desktop GUI, Terminal and Web.

The library is composed by three sub-libraries, each one implements the layer responsible to interpret the Pyforms application on each different environment.

PYFORMS	PYFORMS	PYFORMS
pyforms-gui	pyforms-web	pyforms-terminal

These layers can be user indevidually or toggether if you install pyforms.

Note: Please ★ Star the project at the Github repository to support the project.

CHAPTER 1

Developer

Ricardo Ribeiro	Champalimaud Scientific Software Platform ricardo.ribeiro@research.fchampalimaud.org ricardojvr@gmail.com

1.1 Install & configure

- Install Pyforms using **pip**.

```
pip install pyforms
```

- To install the pyforms layers indevidually check the documentation at:

PYFORMS

pyforms-gui.readthedocs.io

PYFORMS

pyforms-web.readthedocs.io

pyforms-terminal.readthedocs.io

1.1.1 Configure

To switch between execution modes, create the file **local_settings.py** in application running directory with the content:

```
# This flag is used by the module confapp to set these settings as high priority.
SETTINGS_PRIORITY = 0

# The variable is used by pyforms to define the mode it will run.
# It can has the value 'GUI', 'WEB' or 'TERMINAL'.
PYFORMS_MODE = 'TERMINAL'
```

1.2 How it works

On this example it is shown how to create a pyforms application, and how to execute it on the three different environments, the GUI, Web, and Terminal.

1.2.1 Build an app

Create the file **example.py** and add the next code to it.

```
from pyforms.basewidget import BaseWidget
from pyforms.controls   import ControlFile
from pyforms.controls   import ControlText
from pyforms.controls   import ControlSlider
from pyforms.controls   import ControlPlayer
from pyforms.controls   import ControlButton

class ComputerVisionAlgorithm(BaseWidget):

    def __init__(self, *args, **kwargs):
        super().__init__('Computer vision algorithm example')

        #Definition of the forms fields
        self._videofile  = ControlFile('Video')
        self._outputfile = ControlText('Results output file')
        self._threshold  = ControlSlider('Threshold', default=114, minimum=0,␣
↪maximum=255)
        self._blobsize   = ControlSlider('Minimum blob size', default=110,␣
↪minimum=100, maximum=2000)
```

(continues on next page)

(continued from previous page)

```
        self._player     = ControlPlayer('Player')
        self._runbutton  = ControlButton('Run')

        #Define the function that will be called when a file is selected
        self._videofile.changed_event = self.__video_file_selection_event
        #Define the event that will be called when the run button is processed
        self._runbutton.value = self.run_event
        #Define the event called before showing the image in the player
        self._player.process_frame_event = self.__process_frame

        #Define the organization of the Form Controls
        self._formset = [
            ('_videofile', '_outputfile'),
            '_threshold',
            ('_blobsize', '_runbutton'),
            '_player'
        ]


    def __video_file_selection_event(self):
        """
        When the videofile is selected instanciate the video in the player
        """
        self._player.value = self._videofile.value

    def __process_frame(self, frame):
        """
        Do some processing to the frame and return the result frame
        """
        return frame

    def run_event(self):
        """
        After setting the best parameters run the full algorithm
        """
        print("The function was executed", self._videofile.value)


if __name__ == '__main__':

    from pyforms import start_app
    start_app(ComputerVisionAlgorithm)
```

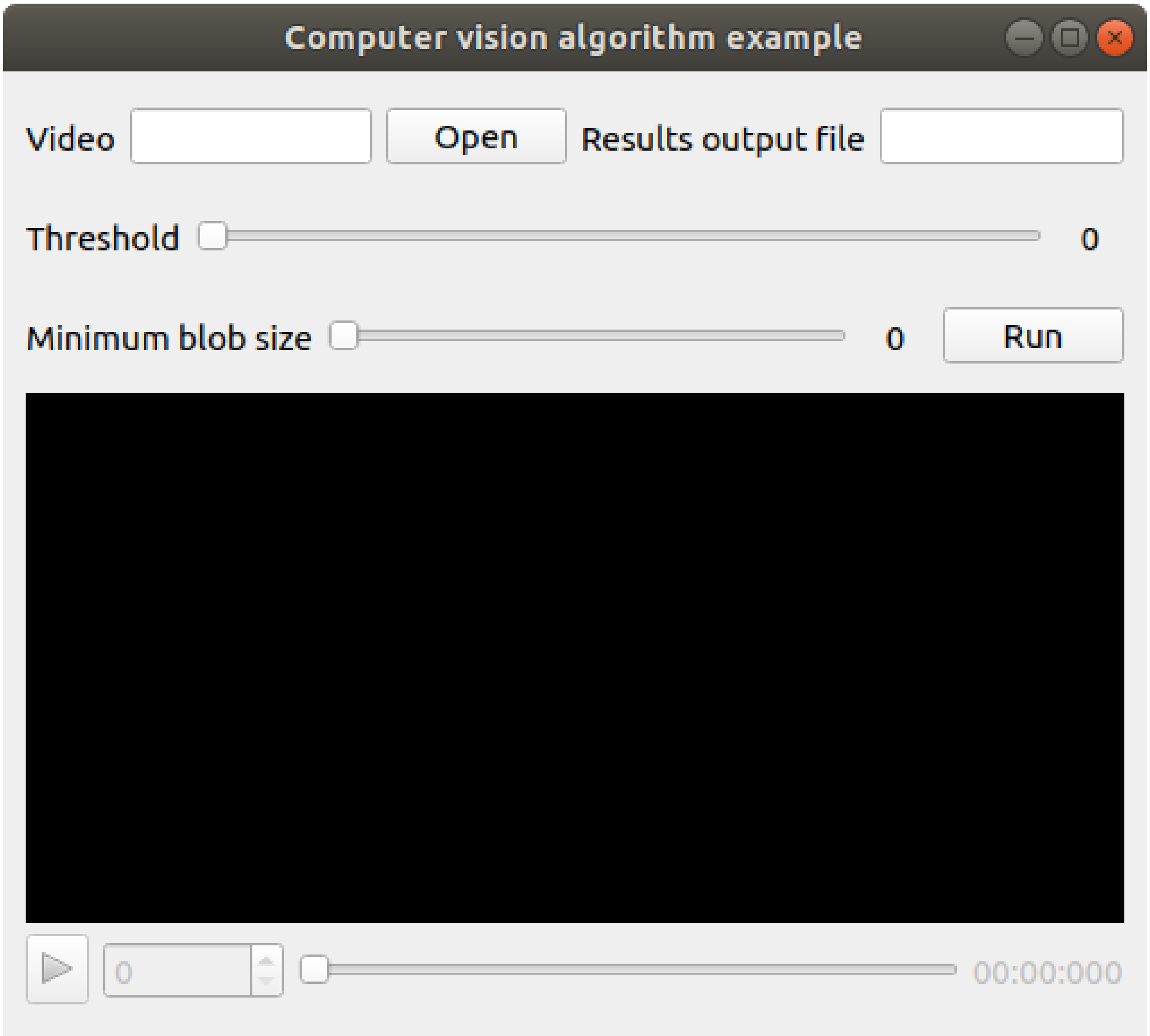
1.2.2 Execute the app in GUI mode

By default the GUI mode is active.

Now execute in the terminal the next command:

```
$ python example.py
```

You will visualize the next window:



1.2.3 Execute the app in TERMINAL mode

Now execute in the terminal the next command:

```
$ python example.py terminal_mode --help
```

```
usage: example.py [-h] [--_videofile _VIDEOFILE] [--_outputfile _OUTPUTFILE]
              [--_threshold _THRESHOLD] [--_blobsize _BLOBSIZE]
              [--exec EXEC] [--load LOAD]
              terminal_mode

positional arguments:
  terminal_mode         Flag to run pyforms in terminal mode

optional arguments:
  -h, --help            show this help message and exit
  --_videofile _VIDEOFILE
```

(continues on next page)

(continued from previous page)

```
                        Video
  --_outputfile _OUTPUTFILE
                        Results output file
  --_threshold _THRESHOLD
                        Threshold
  --_blobsize _BLOBSIZE
                        Minimum blob size
  --exec EXEC           Function from the application that should be executed.
                        Use | to separate a list of functions.
  --load LOAD           Load a json file containing the pyforms form
                        configuration.
```

Set some parameters and execute the function run_event:

```
$ python example.py terminal_mode --_videofile "/home/ricardo/Documents/
↪pictures4presentations/3dscene.mp4" --exec run_event
```

The output will be:

```
The function was executed /home/ricardo/Documents/pictures4presentations/3dscene.mp4
```

1.2.4 Configure the local_setttings to TERMINAL mode

Create the **local_settings.py** file in the application running directory and set the mode in which the application will run.

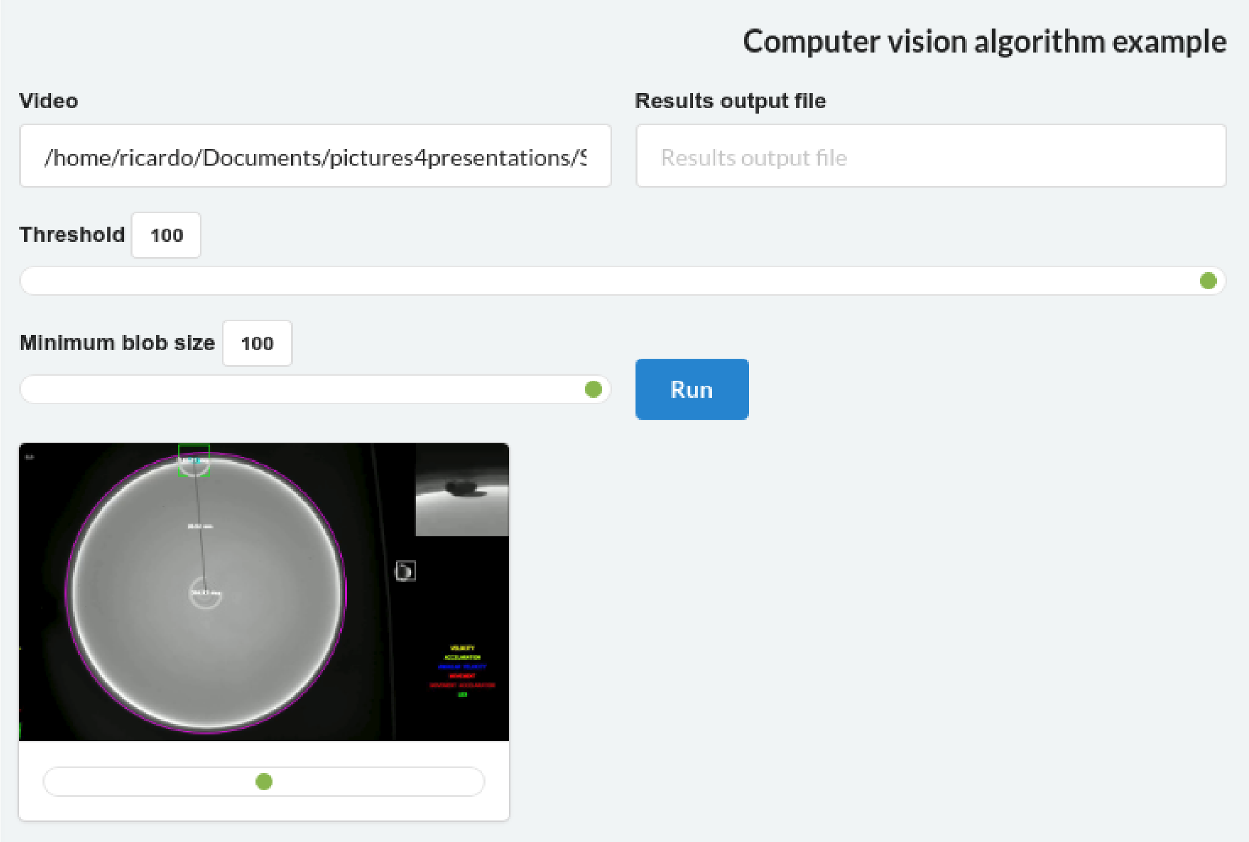
```
# This flag is used by the module confapp to set these settings as high priority.
SETTINGS_PRIORITY = 0

# The variable is used by pyforms to define the mode it will run.
# It can has the value 'GUI', 'WEB' or 'TERMINAL'.
PYFORMS_MODE = 'TERMINAL'
```

You can now run the application in terminal mode without using the parameter **terminal_mode**.

```
$ python example.py --_videofile "/home/ricardo/Documents/pictures4presentations/
↪3dscene.mp4" --exec run_event
```

1.2.5 Execute the app in WEB mode

For information about how to execute the app in WEB mode check the documentation at Pyforms-web@ReadTheDocs.



Computer vision algorithm example

Video

/home/ricardo/Documents/pictures4presentations/S

Results output file

Results output file

Threshold 100

Minimum blob size 100

CHAPTER 2

Indices and tables

- genindex
- modindex
- search