Revise and improve samples to make them more useful and easier to run.

- Author: Vadim Pisarevsky
- Link: The feature request
- Status: **Draft**
- Platforms: **All**
- Complexity: 1-2 man-months

Introduction and Rationale

OpenCV includes several dozens of samples split by multiple directories (e.g. opencv samples are placed in opencv/samples, whereas opencv_contrib samples are put into the subdirectories of the corresponding modules. In general, the samples is very useful part of OpenCV and it really helps our users to start playing with the library. On the other hand, there are some issues:

- Some of the samples have been created long time ago and are very primitive. E.g. morphology.cpp or laplace.cpp or convexhull.cpp. Some of those samples should better be turned into the code snippets in documentation.
- Some other samples have been created recently and largely duplicate each other (e.g. many similar object detection samples in opencv/samples/dnn.
- Some more samples do not really do what they should. For example, letter_recog.cpp sample simply reads the file with precomputed features of each letter, whereas some people may expect that it does live recognition of letters shown to camera (and basically we have all the technology ready for that).
- Quite a few samples need some images and models as input parameters, therefore if you build and run them from Visual Studio, for example, they will likely immediately exit; one need to apply extra efforts to make them work properly.
- Some samples can be controlled by hot keys. This console window is not always displayed and not always visible. Besides, you need to change focus from the console to the highgui window in order to make those hot keys work.
- Another drawback of the samples is that very few of them can run autonomously and do some useful thing. In other words, it's basically impossible to re-use samples as the smoke tests and quickly verify before releases whether they work well (or run at all) or not.

Proposed solution

There are many issues mentioned above and we basically propose to solve the issues one by one.

1. go through the list of samples and maybe remove some obsolete stuff. Some of the samples can be turned into the code snippets. Probably, many of the samples, if not all, can be moved into the corresponding modules.
2. improve some other samples based on the latest technology that we have. The classical examples are digits.py and letter_recog.cpp.
3. make sure that the samples find (or at least try to find) the proper data when they are launched from the file manager without any parameters. For example, OpenCV can provide some function and/or some "path" macro in cvconfig.h that will help to locate the OpenCV source tree and the corresponding samples/data subdirectory.
4. it would be useful to attach some scrolling graphically-rendered "console" window to the highgui windows where the app description and the hot key descriptions could be displayed. Need to see how difficult it would be to implement, probably not very difficult.
5. it would be nice to make the samples scriptable, or maybe use external tool to run the samples and then pass some key presses and mouse events to the highgui windows.

Impact on existing code, compatibility

No any impact on the code, it's only about the samples and the documentation. Some features may require highgui extensions.

Possible alternatives

this proposal includes multiple semi-independent items. We can do some of them, we can postpone or even reject others.

References

TBD