

Maintaining OpenCV 2.4.x

- Author: Vadim Pisarevsky
- Link: The feature request
- Status: **Draft**
- Platforms: All
- Complexity: a few man-weeks per year

Introduction and Rationale

This is intended to be a continuously updated meta-proposal, gathering information on what to do with OpenCV 2.4.x, which is in the maintenance mode for a few years already (since OpenCV 3.0).

On the other hand, OpenCV 2.4.x is still used by many people, that's why it's important to maintain it for some more time, as long as the required effort is reasonably low.

Proposed solution

- Keep 2.4 branch and review/accept PRs coming to it. For now (2018 Spring) we plan to do it for at least 1 more year. Later on, there can be some volunteers engaged to review the patches.
- Define the period when 2.4 will reach the end-of-life and what it will mean for us. One possible scenario is that at some point OpenCV core team will stop creating backports of various patches for 2.4. But we can still review and integrate patches from users, because it's low traffic and correspondingly low cost for us. The good moment is OpenCV 4.0 release.
- One question to discuss is whether we should do 1-time or regular synchronization of some files in OpenCV 3.x and OpenCV 2.4.x, such as video I/O code etc. Usually, it is this non-algorithmic code that needs backports in order to support new OSes, updated APIs etc. This synchronization could simplify propagation of fixes to all OpenCV versions.

Impact on existing code, compatibility

OpenCV 2.4.x support fades slowly, so there should be no big surprises for users. As long as we keep 2.4 branch and accept patches to it, our users should be fine with it.

Possible alternatives

OpenCV 2.4.x decline is happening, so there is no real alternatives, there are nuances in the deprecation process.

References

1. OpenCV 2.4.x branch