

## Planning OpenCV 4.0

- Author: Vadim Pisarevsky
- Link: [The feature request](#)
- Status: **Draft**
- Platforms: **All**
- Complexity: many man-weeks

## Introduction and Rationale

Given that [OpenCV 3.0](#) came out 6 years later after OpenCV 2.0, one may wonder why release 4.0 in “just” 3 years after 3.0 (assuming that 4.0 will be released in 2018, July).

Here are some fundamental reasons (not sorted by importance):

- Switch to C++ 11. OpenCV 3.0 is C++ 98 library, whereas many C++ developers switched to C++ 11 or even later standards. We finally need to follow this route. Migration to C++ 11 means breaking “binary compatibility promise”, so this move alone would require OpenCV 4.0.
- Disclaiming binary compatibility and moving to source-level compatibility. Several years ago we stated that OpenCV will stay binary-compatible between major releases. Technically, it was a necessary thing for OpenCV Manager for Android, but it also was seemingly useful thing for binary OpenCV packages for Linux. Apparently, neither of these two is that important. App developers do not rely on independently-updated package and Linux distro maintainers prefer to do just some security updates. For them the difference between, say, 3.0 and 3.1 is too radical. So we can have more freedom and yet provide a reasonable level of compatibility by relaxing the “binary compatibility” claim to “source compatibility”.
- (mini-)Halide as a part of OpenCV.
- Big revision of functionality and modules content. It's deep learning era now and we need to react.
- Compact OpenCV. The library became quite heavy, we need to strip it down and probably JIT-compile some kernels instead of having them all pre-compiled and wasting space. Stripping down equals broken compatibility, so we will need a major version number increase.
- Possibly big revision of DNN module. It grew fast during 2016, 2017 and 2018. We need to do some refactoring and do not want to be limited too much by the compatibility aspects.

## Proposed solution

OpenCV 4.0 is planned for release in 2018 July. Everything that is planned to be done by 4.0 can be found with a query to our [bug tracker](#). Some feature requests are special - they mirror [the evolution proposals](#). You may find [the evolution proposals targeted for 4.0](#) with another query.

Some of the items may be postponed till some later OpenCV 4.x.

## Impact on existing code, compatibility

Because of the broken compatibility some code may need tweaking to support OpenCV 4.0. Otherwise, there will be [OpenCV 3.x branch](#) that people can use.

## Possible alternatives

Sooner or later, we will need to release the new major OpenCV version. Because the library evolves fast and because we now have enough resources, it's very good moment to do it during 2018.

So, the alternative to keep developing OpenCV 3.x is not that attractive.

**References**

TBD