

[ABOUT](#)[NEWS](#)[EVENTS](#)[RELEASES](#)[PLATFORMS](#)[BOOKS](#)[LINKS](#)[LICENSE](#)

About

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding [14 million](#). The library is used extensively in companies, research groups and by governmental bodies.

Along with well-established companies like Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota that employ the library, there are many startups such as Applied Minds, VideoSurf, and Zeitera, that make extensive use of OpenCV. OpenCV's deployed uses span the range from stitching streetview images together, detecting intrusions in surveillance video in Israel, monitoring mine equipment in China, helping robots navigate and pick up objects at Willow Garage, detection of swimming pool drowning accidents in Europe, running interactive art in Spain and New York, checking runways for debris in Turkey, inspecting labels on products in factories around the world on to rapid face detection in Japan.

It has C++, Python, Java and MATLAB interfaces and supports Windows, Linux, [Android](#) and Mac OS. OpenCV leans mostly towards real-time vision applications and takes advantage of MMX and SSE instructions when available. A full-featured [CUDA](#) and [OpenCL](#) interfaces are being actively developed right now. There are over 500 algorithms and about 10 times as many

functions that compose or support those algorithms. OpenCV is written natively in C++ and has a templated interface that works seamlessly with STL containers.

Support

When something fails

If you are attempting to debug an OpenCV program:

- At first try to troubleshoot the problem using [documentation](#) and [tutorials](#).
- If it doesn't help, search for an answer or ask a question at [OpenCV Answers](#).
- If you found a bug or wish to make a feature request, please see the next section.

Reporting issues / requesting features

- First, check the [issue tracker](#): known bugs, often with patches or workarounds, are generally found there. If you have something to add to an existing bug, add it as a comment to the ticket, rather than posting to the mailing lists.
- Then check the [OpenCV Answers](#) to see if someone else has asked your question or reported your bug.
- If all the above steps failed, the best thing to do is to raise a [ticket](#).

If you're not sure what you found is a bug you can ask on [OpenCV Answers](#). If the problem is confirmed as a bug, please create a ticket.

Asking a question

Please read before posting to any of OpenCV support forums/lists!

Do Not:

- The following are **not** appropriate questions:

- general debugging/programming questions
- questions about software that is not related to OpenCV
- asking for help with your homework
- Do **not** contact the developers/maintainers directly:
 - the community can't see question or answer(s) not asked/answered publicly
 - open source development works best when the entire community participates in discussions and helps to answer questions

Do:

- Be as specific as possible, with steps to reproduce. If we can reproduce the problem, we can fix it quickly.
- Send all questions to [OpenCV Answers](#) or other corresponding mailing list, and report all bugs to the bug tracker.
- Please specify your platform (Windows 32/64bit, Linux x86/x64/ppc/..., MacOSX 32/64bit/ppc); compiler version; OpenCV version/revision; whether IPP, OpenMP, MMX, SSE ... have been enabled or not, and any other information that let us reproduce the environment, identify and localize the problem.
- A sample code reproducing the problem helps us the most. Please check [samples](#) - for the preferred style (small code size, cross-platform). There is **no need** to provide any project files or makefiles, if it is a short single-file sample, we could build it and run in a minute.
- Please provide the output of `cv::getBuildInformation()` function every time you report a bug about the trunk version of OpenCV. The problem can be in your build settings, so the log can greatly help us.
- If your application is complex, and the problem happens somewhere in the middle, often it is still possible to create a short standalone sample:
 - capture and store the data that you pass to the function using `FileStorage`
 - copy the function call and put the corresponding reading from `FileStorage` in front of it
- Describe exactly what you were doing or are trying to do and what went wrong. If you say: *"OpenCV doesn't work!"* - we will not be able to help you.
- As appropriate, also include your:
 - backtraces
 - relevant config files

- screenshots or videos to demonstrate the problem

Contribute

If you want to help us, there are many ways to do it. By the way, did you see the “*Donate*” button on the main page? :)

Already have some code that you want to see as a part of the library? Please follow [this guide](#).

Users can:

- Help others, fill out some answers on our [Q&A forum](#). You don't have to be a guru to help.
- Write cool tutorials: please follow the [instructions](#).
- Submit cool demo videos to our [YouTube channel](#).
- Say “thanks” with [ratings and comments](#).

Developers can:

- Participate in discussions regarding [new code submissions](#). We have a lot of pull requests, and we need more people to review and test them! Actually you may become an official reviewer in the OpenCV project one day.
- If you need a simple task to start from, please consider [these ones](#). They are simple enough, so you can get familiar with the [contribution process](#).
- Contribute code: it has to be [BSD](#) and should not add extra dependencies. More information about a code submission is available on our developer site: [contribution guide](#) and [coding style guide](#).
- Test our latest code from the [Git repository](#) and report/fix any bugs you find. [Reporting bugs](#).
- Help us optimize the bindings for different platforms/languages: CUDA, NEON, Python, Android or iOS.
- Help us improve the unit tests, documentation, samples.

Company can:

- Become a sponsor to help hire developers and organize events.

- Contribute to the OpenCV library by providing coder time or by being part of development decisions.
- Provide the infrastructure (testing farm, build farm, website).

Contacts

Found a typo or have any suggestions regarding this site? Send us an [email](#).

If you have any other questions, feel free to contact us by [email](#).

 [E-Mail](#)

 [Slack](#)

 [GitHub](#)

 [Facebook](#)

 [Google+](#)

 [Wikipedia](#)

 [Forum](#)

 [IRC #opencv](#)

 [SourceForge](#)

 [Twitter](#)

 [YouTube](#)

© Copyright 2018, OpenCV team