

16623 - Advanced Computer Vision Apps

Final Project (50 % of total grade)

Project Proposal Due - Sunday the 6th of November

Project CheckPoint Due - Thursday the 17th of November

Final Project Due - Friday the 8th of December

Your 16-623 final project gives you the opportunity to dive deeply into a mobile computer vision problem of your choosing for the final month of the course. Perhaps of more importance, is the opportunity for eternal fame and glory of winning 16623's inaugural "best project" competition (prize to be determined). All projects are expected to be in the form of an App that can be built and run on the iOS device provided to you in class. What you attempt for your project, however, is completely up to you.

Project Proposal (due Sun, November 6th)

The purpose of the proposal is two fold:

1. Writing your ideas down forces you to organize your thoughts about the project.
2. It gives us (instructors/TAs) the ability to verify your plans are of the right scope (also offers the ability to offer suggestions and help).

Please create a public GitHub page for your project (to do this you will need to have or create your own GitHub account). Then, to tell us about your project, please fill out the following form <http://goo.gl/forms/rfjW6X9ypg>.

Your project GitHub page should contain the following sections and content (simply modify the README.md file) .

Title: Please provide the title of your project, followed by the names of all team members. Teams may include up to two students (special permission can be obtained for a team of three although there must be a very good reason).

Summary: Summarize your project in no more than 2-3 sentences. Describe what you plan to do and what mobile vision problem you will be working on. See 16623.courses.cs.cmu.edu/ideas for example project ideas.

Background: If your project involves taking advantage of computational speedups available on your iOS device - such as box filters, inverse composition in the LK algorithm, NEON intrinsics, OpenGL ES, Accelerate Framework, binary descriptors such as FAST and BRIEF, etc. - describe their application and why they are necessary in more detail. If your project involves something around using your device in a mobile fashion - for example virtually rendering an object in your room - then describe what components of your solution are unique to a mobile device (the high-speed camera, GPS, IMU, Gyro?).

The Challenge: Describe in a few sentences why the problem is challenging. Could you solve your problem using just a few pre-existing functions in OpenCV? Try to state explicitly what you are hoping to learn by doing this project and how is it related to mobile computer vision? A flow chart or visual depiction of what you are trying to do would be good here. Some questions to consider:

- Does the challenge lie in making an existing vision algorithm more efficient in iOS?
- Are you trying to solve a real-world problem that can only be solved through a combination of computer vision and mobile computing?

Goals & Deliverables: Describe the deliverables or goals of your project.

- In a couple of sentences separate your goals into what you **PLAN TO ACHIEVE** (what you believe you must get done to have a successful project and get the grade you expect) and an extra goal or two that you **HOPE TO ACHIEVE** if the project goes really well and you get ahead of schedule.
- Describe what success looks like and how it can be evaluated. For example, if your project is to measure the velocity of a baseball being thrown in front of an iOS device, how will you validate that it works? Screen shots of the App in action? A speed benchmark run across a variety of videos? A live video of the app in action? It will NOT be enough to simply provide the Xcode project - you will need to provide evidence that you have achieved your goal.
- How realistic is it for your team to get what it needs to get done within the allotted time? Remember you only have a few weeks to get this project completed.

Schedule: Produce a schedule for your project. Your schedule should have at least one item to do per week per participant. List what your plan to get done each week from now until the 11th of December deadline.

Project CheckPoint (due Wednesday, November 16th midnight EST)

The checkpoint exists is to give you a deadline approximately halfway through the project. The following are suggestions for information to include in your **checkpoint write-up**. Your goal in the writeup is to assure the course staff (and yourself) that your project is proceeding as you said it would in your proposal. If it is not, your checkpoint writeup should emphasize what has been causing you problems, and provide an adjusted schedule and adjusted goals. **Important:** Submit a pdf of your project checkpoint writeup (called **CheckPoint.pdf**) to your your AFS dropbox /afs/cs.cmu.edu/academic/class/16623-f16-users/andrew_id.

- Make sure your project schedule on your GitHub project page is up to date with work completed so far, and well as with a revised plan of work for the coming weeks. As by this time you should have a good understanding of what is required to complete your project, I want to see a very detailed schedule for the coming weeks. I suggest breaking time down into half-week increments. Each increment should have at least one task, and for each task put a person's name on it.
- Make sure you are regularly committing changes and updates to your GitHub project page. We want to see that everyone is contributing equally. **Try to be descriptive when you make your commits so it is clear what you are doing and what changes you have been making.**
- One to two paragraphs, summarize the work that you have completed so far.
- Describe how you are doing with respect to the goals and deliverables stated in your proposal. Do you still believe you will be able to produce all your deliverables? If not, why? What about the "nice to haves"? In your checkpoint writeup we want a new list of goals that you plan to hit.
- What do you plan to show? Remember you need to provide a YouTube clip describing your app in class during the project presentation - so something visual will be important.
- **Do you have any preliminary results at this time?** Would be great to post some of these (either images, links to videos, tables or graphs).
- (Optionally) Do you need to schedule a meeting with 16623 staff to discuss any issues?

Project Presentation (due Thu, December 8th)

Each team will be given approximately 2.5 minutes per member to present (for example a 2 member team will have 5 minutes allotted). Each team will fill out the following form, providing a short (must be shorter than your allotted time) YouTube clip describing your App in action. Teams can submit their YouTube clips through the form <http://goo.gl/forms/YoeQt0c1Hf>. 16623 staff will select the 3 best presentations, with the winner receiving the the best project prize.

Final Write Up (due Fri, December 11th)

Your final writeup should include the following basic sections. You are also encouraged to provide more detail if you wish. Note that some of the information in your final writeup can be pulled directly from your proposal if it is still accurate. We are looking for the project write up to be a CVPR style document of at least around 1-2 pages per participant in the team (i.e. a team of 2 would need to produce at least 3-4 page document). Style files for CVPR (in Latex or Word) can be found here:- http://www.pamitc.org/cvpr15/author_guidelines.php. **Important:** Submit a pdf of your final project writeup (called `FinalWriteUp.pdf`) to your your AFS dropbox `/afs/cs.cmu.edu/academic/class/16623-f16-users/andrew_id`.

Abstract: A short (no more than a paragraph) project summary. If applicable, the summary should list your project deliverables and why your problem is challenging when it comes to mobile computer vision.

Introduction: Describe the algorithm, application or system you are implementing into a computer vision app. Why is it a good candidate for being placed into a vision app. Why is it challenging? A good question to ask always here is, why can't this project be trivially implemented in OpenCV in just a few lines of code?

Background: Please cite relevant papers, software packages, etc. that relate to your problem. Ensure you describe in at least a paragraph or how this background material directly relates to your chosen problem.

Approach: Tell us how your implementation works and why it is different. Please include enough math here to explain your approach succinctly. Your description should be sufficiently detailed to provide the course staff a basic understanding of your approach. Try to make a good argument for why you have chosen your solution. Further, attempt to clarify at what level of mobile vision your solution lies: (i) algorithmic, (ii) software, (iii) architecture, or (iv) SOC hardware - recall notes from Lecture 2 for a refresher on these levels.

Results: Please present the results of your App in action. This may take the form of screen shots, images, graphs, tables etc. The focus of the results section should be to show the utility of your approach over a naive implementation. If your project is centered around making a vision algorithm more efficient in iOS I would like you to try to use a tool in Xcode called "Instruments" (see a tutorial on its use here <http://www.raywenderlich.com/23037/how-to-use-instruments-in-xcode>). In particular, screen grabs from Instruments showing what parts of your code are consuming the most resources would be useful - it is also useful for working out where memory leaks and other errors are occurring. Before (i.e. naive implementation) versus after (i.e optimized) comparisons will also be useful to make your case.

List of Work: If your project is a team project, please list the work performed by each partner. Alternatively, you can simply state: "equal work was performed by both project members."

GitHub Page: Place the link to your GitHub page in the document, also please ensure that the page is updated with all the final results. Describe in a sentence of two what changes you made to your GitHub page from the project checkpoint.

References: A list of references that you are citing in the rest of the document.