



(<http://vision.stanford.edu/>) (<http://stanford.edu/>)



# CS231n: Convolutional Neural Networks for Visual Recognition

([index.html](#))

## Course Project

### Previous Projects

Update: Winter Quater 2016 projects ([reports.html](#)) have now been posted!

### Overview

The Course Project is an opportunity for you to apply what you have learned in class to a problem of your interest. There are two project options you can pick from:

#### Option 1: Your own project (Encouraged)

---

You are encouraged to select a topic and work on your own project. Potential projects usually fall into these two tracks:

- **Applications.** If you're coming to the class with a specific background and interests (e.g. biology, engineering, physics), we'd love to see you apply ConvNets to problems related to your particular domain of interest. Pick a real-world problem and apply ConvNets to solve it.
- **Models.** You can build a new model (algorithm) with ConvNets, or a new variant of existing models, and apply it to tackle vision tasks. This track might be more challenging, and sometimes leads to a piece of publishable work.

One **restriction** to note is that this is a Computer Vision class, so your project should involve pixels of visual data in some form somewhere. E.g. a pure NLP project is not a good choice, even if your approach involves ConvNets.

To inspire ideas, you might look at recent deep learning publications from top-tier vision conferences, as well as other resources below.

- Awesome Deep Vision (<https://github.com/kjw0612/awesome-deep-vision>)
- CVPR ([http://www.pamitc.org/cvpr14/accepted\\_papers.php](http://www.pamitc.org/cvpr14/accepted_papers.php)): IEEE Conference on Computer Vision and Pattern Recognition
- ICCV (<http://www.cvpapers.com/iccv2013.html>): International Conference on Computer Vision
- ECCV (<http://eccv2014.org/accepted-papers/>): European Conference on Computer Vision
- NIPS (<http://nips.cc/Conferences/2014/Program/accepted-papers.php>): Neural Information Processing Systems
- ICLR (<http://openreview.net/venue/iclr2014>): International Conference on Learning Representations
- Past CS229 Projects (<http://cs229.stanford.edu/projects2013.html>): Example projects from Stanford machine learning class
- Kaggle challenges (<http://www.kaggle.com/>): An online machine learning competition website. For example, a Yelp classification challenge (<https://www.kaggle.com/c/yelp-restaurant-photo-classification>).

For applications, this type of projects would involve careful data preparation, an appropriate loss function, details of training and cross-validation and good test set evaluations and model comparisons. Don't be afraid to think outside of the box. Some successful examples can be found below:

- Teaching Deep Convolutional Neural Networks to Play Go (<http://arxiv.org/abs/1412.3409>)
- Playing Atari with Deep Reinforcement Learning (<http://arxiv.org/abs/1312.5602>)
- Winning the Galaxy Challenge with convnets (<http://blog.kaggle.com/2014/04/18/winning-the-galaxy-challenge-with-convnets/>)

ConvNets also run in real time on mobile phones and Raspberry Pi's - feel free to go the embedded way. You may find DeepBeliefSDK (<https://github.com/jetpacapp/DeepBeliefSDK>) helpful. This particular project might be slightly out of date, but it may help you find more like it.

For models, ConvNets have been successfully used in a variety of computer vision tasks. This type of projects would involve understanding the state-of-the-art vision models, and building new models or improving existing models for a vision task. The list below presents some papers on recent advances of ConvNets in the computer vision community.

- **Object recognition:** [Krizhevsky et al.] (<http://www.cs.toronto.edu/~fritz/absps/imagenet.pdf>), [Russakovsky et al.] (<http://arxiv.org/abs/1409.0575>), [Szegedy et al.] (<http://arxiv.org/abs/1409.4842>), [Simonyan et al.] (<http://arxiv.org/abs/1409.1556>), [He et al.] (<http://arxiv.org/abs/1406.4729>)
- **Object detection:** [Girshick et al.] (<http://arxiv.org/abs/1311.2524>), [Sermanet et al.] (<http://arxiv.org/abs/1312.6229>), [Erhan et al.] (<http://arxiv.org/abs/1312.2249>)

- **Image segmentation:** [Long et al.] (<http://arxiv.org/abs/1411.4038>)
- **Video classification:** [Karpathy et al.] (<http://cs.stanford.edu/people/karpathy/deepvideo/>), [Simonyan and Zisserman] (<http://arxiv.org/abs/1406.2199>)
- **Scene classification:** [Zhou et al.] (<http://places.csail.mit.edu/>)
- **Face recognition:** [Taigman et al.] ([http://www.cv-foundation.org/openaccess/content\\_cvpr\\_2014/papers/Taigman\\_DeepFace\\_Closing\\_the\\_2014\\_CVPR\\_paper.pdf](http://www.cv-foundation.org/openaccess/content_cvpr_2014/papers/Taigman_DeepFace_Closing_the_2014_CVPR_paper.pdf))
- **Depth estimation:** [Eigen et al.] (<http://www.cs.nyu.edu/~deigen/depth/>)
- **Image-to-sentence generation:** [Karpathy and Fei-Fei] (<http://cs.stanford.edu/people/karpathy/deepimagesent/>), [Donahue et al.] (<http://arxiv.org/abs/1411.4389>), [Vinyals et al.] (<http://arxiv.org/abs/1411.4555>)
- **Visualization and optimization:** [Szegedy et al.] (<http://arxiv.org/pdf/1312.6199v4.pdf>), [Nguyen et al.] (<http://arxiv.org/abs/1412.1897>), [Zeiler and Fergus] (<http://arxiv.org/abs/1311.2901>), [Goodfellow et al.] (<http://arxiv.org/abs/1412.6572>), [Schaul et al.] (<http://arxiv.org/abs/1312.6055>)

You are welcome to come to our office hours to brainstorm and suggest your project ideas. We also provide a list of popular computer vision datasets:

- Meta Pointer: A large collection organized by CV Datasets. (<http://www.cvpapers.com/datasets.html>)
- Yet another Meta pointer (<http://riemenschneider.hayko.at/vision/dataset/>)
- ImageNet (<http://http://image-net.org/>): a large-scale image dataset for visual recognition organized by WordNet (<http://wordnet.princeton.edu/>) hierarchy
- SUN Database (<http://groups.csail.mit.edu/vision/SUN/>): a benchmark for scene recognition and object detection with annotated scene categories and segmented objects
- Places Database (<http://places.csail.mit.edu/>): a scene-centric database with 205 scene categories and 2.5 millions of labelled images
- NYU Depth Dataset v2 ([http://cs.nyu.edu/~silberman/datasets/nyu\\_depth\\_v2.html](http://cs.nyu.edu/~silberman/datasets/nyu_depth_v2.html)): a RGB-D dataset of segmented indoor scenes
- Microsoft COCO (<http://mscoco.org/>): a new benchmark for image recognition, segmentation and captioning
- Flickr100M (<http://yahoolabs.tumblr.com/post/89783581601/one-hundred-million-creative-commons-flickr-images>): 100 million creative commons Flickr images
- Labeled Faces in the Wild (<http://vis-www.cs.umass.edu/lfw/>): a dataset of 13,000 labeled face photographs
- Human Pose Dataset (<http://human-pose.mpi-inf.mpg.de/>): a benchmark for articulated human pose estimation
- YouTube Faces DB (<http://www.cs.tau.ac.il/~wolf/ytfaces/>): a face video dataset for unconstrained face recognition in videos
- UCF101 (<http://crcv.ucf.edu/data/UCF101.php>): an action recognition data set of realistic action videos with 101 action categories
- HMDB-51 (<http://serre-lab.clps.brown.edu/resource/hmdb-a-large-human-motion-database/>): a large human motion dataset of 51 action classes

#### Option 2: Tiny ImageNet Challenge

---

If you are unable to come up with a project idea, you can fall back to working on the Tiny ImageNet Challenge which we will run similar to the ImageNet challenge. The goal of the challenge will be for you to do as well as possible on the Image Classification problem. You will submit your final predictions on a test set to our evaluation server and we will maintain a class leaderboard.

- Tiny ImageNet dataset (<http://cs231n.stanford.edu/tiny-imagenet-200.zip>)
- Evaluation server (<https://tiny-imagenet.herokuapp.com/>)

## Important Dates

Course project proposal: due January 30.

Course project milestone: due February 17.

The potter session will be held 2-5pm at Gates (AT&T patio) on March 9.

Final course project: due **March 13** (11:59pm).

## Grading Policy

```

Final Project: 40%
milestone: 5%
write-up: 10%
  • clarity, structure, language, references: 3%
  • background literature survey, good understanding of the problem: 3%
  • good insights and discussions of methodology, analysis, results, etc.: 4%
technical: 12%
  • correctness: 4%
  • depth: 4%
  • innovation: 4%
evaluation and results: 10%
  • sound evaluation metric: 3%
  • thoroughness in analysis and experimentation: 3%
  • results and performance: 4%
poster: 3% (+2% bonus for best few posters)

```

## Project Proposal

The project proposal should be one paragraph (200-400 words). If you work on your own project, your proposal should contain:

- What is the problem that you will be investigating? Why is it interesting?
- What data will you use? If you are collecting new datasets, how do you plan to collect them?
- What method or algorithm are you proposing? If there are existing implementations, will you use them and how? How do you plan to improve or modify such implementations?
- What reading will you examine to provide context and background?
- How will you evaluate your results? Qualitatively, what kind of results do you expect (e.g. plots or figures)? Quantitatively, what kind of analysis will you use to evaluate and/or compare your results (e.g. what performance metrics or statistical tests)?

If you choose to work on Tiny ImageNet Challenge, emphasize the last three bullet points on the list above. Each group should send a plain-text email (subject: **[cs231n] project proposal + your SUNet IDs**) of your project proposal to cs231n-winter1516-staff@lists.stanford.edu (mailto:cs231n-winter1516-staff@lists.stanford.edu). Don't attach any files (such as Word, PDF, etc.). Please have your teammate cc'ed if any. If your proposed project is joint with another class' project (with the consent of the other class' instructor), make this clear in the proposal.

## Project Milestone

Your project milestone report should be between 2 - 3 pages using the provided template (<http://www.pamitc.org/cvpr15/files/cvpr2015AuthorKit.zip>). The following is a suggested structure for your report:

- Title, Author(s)
- Introduction: this section introduces your problem, and the overall plan for approaching your problem
- Problem statement: Describe your problem precisely specifying the dataset to be used, expected results and evaluation
- Technical Approach: Describe the methods you intend to apply to solve the given problem
- Intermediate/Preliminary Results: State and evaluate your results upto the milestone

**Submission:** Please upload a PDF file named **<your SUNet ID>\_milestone.pdf** to the assignments tab on coursework (<https://coursework.stanford.edu>). Note that, each individual in a team is required to make submission (i.e. the same PDF) for grading purpose. The late days are counted by the timestamp of the last submission in the team.

## Final Submission

Your final write-up should be between **6 - 8** pages using the provided template (<http://www.pamitc.org/cvpr15/files/cvpr2015AuthorKit.zip>). After the class, we will post all the final reports online so that you can read about each others' work. If you do not want your writeup to be posted online, then please let us know at least a week in advance of the final writeup submission deadline.

Submit your final submission through **CourseWork**. You will submit one or two files:

1. A pdf file of your final report
2. (OPTIONAL) zip file (or pdf file) with Supplementary Materials

**Report.** The following is a suggested structure for the report:

- Title, Author(s)
- Abstract: It should not be more than 300 words
- Introduction: this section introduces your problem, and the overall plan for approaching your problem
- Background/Related Work: This section discusses relevant literature for your project
- Approach: This section details the framework of your project. Be specific, which means you might want to include equations, figures, plots, etc

- Experiment: This section begins with what kind of experiments you're doing, what kind of dataset(s) you're using, and what is the way you measure or evaluate your results. It then shows in details the results of your experiments. By details, we mean both quantitative evaluations (show numbers, figures, tables, etc) as well as qualitative results (show images, example results, etc).
- Conclusion: What have you learned? Suggest future ideas.
- References: This is absolutely necessary.

**Supplementary Material** is not counted toward your 6-8 page limit.

Examples of things to put in your supplementary material:

- Source code (if your project proposed an algorithm, or code that is relevant and important for your project.).
- Cool videos, interactive visualizations, demos, etc.

Examples of things to not put in your supplementary material:

- All of Caffe source code.
- Various ordinary data preprocessing scripts.
- Any code that is larger than 1MB.
- Model checkpoints.
- A computer virus.

## Poster Session

We will hold a poster session in which you will present the results of your projects in form of a poster. The poster session will happen on March 9th, 2:00-5:00pm, at AT&T patio (the lawn behind Gates building). Poster boards and easels will be provided.

## Example Project Reports

Your project reports should structure like a computer vision conference paper (CVPR, ECCV, ICCV, etc.). You can find publications from Stanford Vision Lab from here (<http://vision.stanford.edu/publications.html>). In addition, you may also take a look at some previous projects from other Stanford CS classes, such as CS221 (<http://web.stanford.edu/class/cs221/sample-projects/>), CS229 (<http://cs229.stanford.edu/projects2013.html>) and CS224W (<http://web.stanford.edu/class/cs224w/projects.html>)

## Collaboration Policy

You can work in teams of up to **3** people (note: in 2015 this was 2, we're changing this to 3 in 2016). We do expect that projects done with 3 people have more impressive writeup and results than projects done with 2 people. To get a sense for the scope and expectations for 2-people projects have a look at project reports from the last year.

## Honor Code

You may consult any papers, books, online references, or publicly available implementations for ideas and code that you may want to incorporate into your strategy or algorithm, so long as you clearly cite your sources in your code and your writeup. However, under no circumstances may you look at another group's code or incorporate their code into your project.

If you are doing a similar project for another class, you must make this clear and write down the exact portion of the project that is being counted for CS231n.