I think this is a very interesting project! Please find my feedback below. Please let me know if you have any questions on this or if anything is unclear.

What I like about this proposal:

I like that the project has clear applications, both for colorizing old photographs and movies, but also for colorizing images that usually are not colored even today, such as medical and astronomical images.

I also like that it is easy to see and understand the output of the algorithm: If the colorization looks good, then the algorithm works!

What I would have done differently:

I am not sure if I would have done anything differently. However, here are some of the thoughts and questions I have after reading the proposal.

- Is there a difference between colorizing an image and a movie? The simplest idea seems to be to simply colorize a movie by going through it frame by frame and colorize each frame as if it was an image. But is it possible to use the fact that the frames come from the same movie to improve on this? For example, knowing that the frames come in sequence after one another—often with small changes between subsequent frames—is it possible to improve colorization? Or would you use a totally different techniques for colorizing images and videos?
- It is very interesting how this might be applicable to medical imaging, and other kinds of imaging which are not colored. However, will a machine learning model which is trained on, say, the kind of images available in the Berkeley Segmentation Dataset generalize to e.g. MRI scans or satellite imagery? Or would you need to train different machine learning models using data from each particular application area separately?
- For the datasets, it seems like the Berkeley Segmentation Dataset already has colored and gray scale images. Would you simply convert the color images from ImageNet and CVCL to gray scale manually?
- You mention supervised learning methods such as Linear Logistic Regression and SVM. How can these classification methods be applied to colorization? Since each pixel in a regular color image is represented by an RGB triplet (R,G,B), with each coordinate taking integer values in [0, 255], wouldn't this mean you would have to use a very large number of classes? Would regular regression be more appropriate here?

What I wish that this project can achieve with unlimited resources:

It would be quite cool if you got to the point where you could do movie colorization. It also would be very interesting to see how well a model like this can do on e.g. medical images and astronomical images.

I look forward to seeing the final results of this project!

First off, I really like your topic. It's a cool image processing problem which is hard to automate, so probably a good field in which to try and apply ML methods.

What I like:

• I like that you have a good data set to work with, notably the Berkley one which has test and training data.

• This problem also makes data generation fairly easy: you can write up a simple python script using openCV to turn a image to b/w. So you guys could generate boat loads of data if you want to (for training or testing). Which is kind of neat.

• I also like the impacts and the use of such a tool. For movies to scientific imagery, it has a nice breadth to it.

What I would have done differently:
I don't know what you guys have in mind yet, so this is more of a couple suggestions if you need them:

• Are you going to train on different types of images and test on the same types of images afterwards? It seems like images of landscapes or astronomical images will have pretty different color palets, and that might lead to "mis classified" images.

• For evaluating correctness (I see that you have already put some thought into it), are you going to allow for a certain error margin to say that a pixel was well classified? It seems that the most important thing would be to assign a shade of green for grass, and whether it's the correct shade is really less important. Not sure how to quantify these things, maybe by classifying colors independently of the RGB amount and adding that as a feature?

What I would like the project to achieve with unlimited resources:

• I couldn't put this bullet in the other section because it might be too much work, but it would be interesting to have a first learning algorithm that would try to classify the picture before doing colorization. That way when the picture goes through the colorization algorithm, it has a better chance of getting the right colors. That could be a good way to make the learning algorithm more robust: by giving cleaner, more well understood data from the start.

• Besides that I think your goals are really neat. I'm sure you'll have some interesting results, both on the ML side and visually.

Best of luck, look forward to seeing your results.
-Thibaud
Reply to this followup discussion

Hi folks, cool ideas in this proposal. My feedback is below:

What I like about this proposal

- Image colorization is a neat idea that should make for an engaging presentation at the end of the course. Widely applicable too (see moonshot ideas below).
- It seems like you have a good initial understanding of prior work on this topic.
- You have identified lots of different machine learning approaches you can use; this will make for interesting comparative results and good division of work among group members.

What I would have done differently (practical constructive feedback for improvements)

- Echoing other students' feedback, I think you're on a good track, but I do have some questions and feedback in the next bullet points.
- Some of the sections on machine learning methods are pretty vague, especially the one on CNNs. There are a couple of citations, but I would have liked to see explanations of what each of the cited papers' or websites' intellectual contributions are.
- People have done image colorization before according to your citations -- I would have liked to see a discussion of the pros/cons of the different ML techniques and your expectations of how well they will perform.
- There is no explanation of SIFT/SURF.
- I guess these points reduce to "I would have liked better discussion of the sources and ML approaches." After reading the proposal, I understand the general idea and what approaches exist, but I don't know how to feel about them, and I could not explain how the ML approaches actually work for this application. (The discussion of k-means clustering was good though).
- I assume that given a large enough dataset, lots of different images will have similar features with different true colors. For example, a tree trunk from close-up might have similar edge structure to a zoomed-out picture of an office building (or something). I worry that without some sort of image classification first that tells the algorithm "this is a tree, not an office building, so colorize it using only the training examples that contained trees" the algorithm will get color predictions wrong in some cases.

What I wish that this project can achieve with unlimited resources (moonshot ideas)

- To address the concern about "tree v. office building" above, it could be cool to train one colorization algorithm for "nature" images, one for "city" images, etc. and then see how they perform on within-class test images, and outside-class test images.
- Image colorization involves predicting the values of three related channels using one channel of input. In scientific applications, there are many instances where one field is a good predictor of one or several others. It would be cool to extend this to predicting precipitation intensity (one field) given satellite image of cloud cover, or wind speed and direction (two fields) given the same image. Or some other crazy example from another scientific area! I'm not an expert, but your work sounds broadly applicable.
- It would be fun to colorize an old black-and-white movie clip, especially to see how sensitive each algorithm is to small changes in the image input as characters gradually move around on the screen, etc.
- Overall, this sounds like a cool project and I look forward to seeing your results!

What I like about this proposal

- Enough literature survey has been done
- Features considered are fairly good
- Detailed thought process for each learning approach considered
- Idea of colorizing medical and scientific images

What I would have done differently (practical constructive feedback for improvements)

- Dataset - A dataset generated by converting colorized images to gray-scale would provide a ton of training data and makes life easier while evaluating the results
- Methodology - Using a similar sample image might help in predicting the color better. i.e., for each gray scale image, find a similar image in the training set and for each pixel in the gray image, find the color of the pixel in the similar image which matches the intensity value most closely.
- Evaluation- Instead of going by the closeness of the predicted pixel color and the actual color, we can check if predicted pixel color and actual pixel color belong to the same color family. I feel this provides a better measure of accuracy because a small change in the values of rgb can change the color family. So, even a threshold for the difference in pixels might not help.

What I wish that this project can achieve with unlimited resources (moonshot ideas)

- Given all the time and resources, I think using techniques like Segmentation and Transfer learning would give the best results. Segmentation extracts contours from an image by assigning a label to every pixel such that pixels with the same label share certain characteristics. This technique would help in detecting the objects, parts of objects, boundaries better. This might provide near to accurate results, especially when the image is complicated, like an eye, with lot of closer segments and highly variant neighboring pixels. This would also work well with scientific images where we see lot of unlabeled data.
- Though not important, colorizing random objects like a chair/a pen/a car based on the surrounding objects so that it is distinctly seen. For example, a car with a blue background might better be colored in black than in blue.

Reply to this followup discussion

What I like:
Can I say everything? I am a history nerd and am a big fan of this kind of image restoration. I'm a regular on the subreddit you mentioned in your presentation as well. Reddit's own colorizeBot could definitely use some help :/. As far as the project itself goes, you have a wealth of data available which is good. Prof. Ketelsen even mentioned that any image you take could be converted to gray scale and then used to benchmark your algorithms ability to return it to its original state.

Concerns:
One concern that stood out was your proposed method for testing. I think asking users is overly complex and prone to error. Instead, there are a wealth of images that have already been professionally restored (for example entire reels of footage from the World Wars have been restored to color based on known colors of uniforms, environmental conditions, and even weather). I also agree with the comment made in your presentation that basing accuracy on distance between individual pixel values is better than taking a binary match.

Moonshot:
I think the moonshot for this goal would be to make it publicly available and turn it into an unsupervised or reinforcement learning algorithm that gets better with use. For example. reddit's colorizeBot is pretty bad but with your algorithm, perhaps you could not only replace it with a better version, but replace it with a better version that improves the more images it processes.

What I like:

- There are so many cool and useful applications of colorization in many different areas
- The many different approaches that have already been developed for image colorization. It seems like there has been a lot of work and research done in this field already that will help guide you.
- The huge amount of data that is available from so many different sources. Getting data and reference images will be easy for you, definitely.
- Just the concept of colorizing images seems really interesting, and will be visually fun to play with as you do your project, and be really enjoyable to see demonstrated during the final presentation.

Concerns:

- My main concern would probably be the lack of a narrow focus to your approach. Your proposal has a bunch of different methods, with a bunch of different datasets, using different machine learning techniques, on a wide variety of images from different sources, for different purposes. I would like to see you focus in on a specific method of colorization on a specific type of image for a specific purpose, using specific technique(s). I think defining the scope of your project like that will make it easier and better in the end.
- Build in some concrete way of determining accuracy. Maybe that's converting a color image to grayscale and then comparing the pixel values, or having a reference image. Purely human feedback won't be specific enough, I think.
- You might try a specific subclass or type of images to colorize. For example, trees or cities or ocean, that have a similar color palette, and investigate how optimizing for a type of image effects accuracy.

Moonshot:

- Some sort of image recognition. If you could identify "this is the sky. sky should be blue." or "this is grass, grass should be green.", that would be really cool.
- Incorporating some sort of reverse image search into your algorithm. If you could somehow automatically do a reverse Google image search on a training image, pull in some amount of the colored results, and then train against those similar images in your colorizing algorithm, that could be interesting.