Laura Walsh

Utilising deep learning techniques to colourise greyscale images and videos

A machine learning beginner’s guide

**The Beginning:**

Artificial Intelligence? Machine Learning? Deep Learning? When I arrived at the Infosys campus in Bangalore 9 weeks ago these were merely buzz words I had no experience with. As a computer science student I am comfortable with programming a computer to do what I want it to do, but to make a computer think for itself is a different matter entirely. So, when I was tasked with making a program to convert black and white images and videos to colour I was initially daunted. This is my journey from complete novice to somewhat knowledgeable machine learning enthusiast.

My first port of call was to learn more about the field of Artificial Intelligence and get an understanding of where the idea came from and where we were today. In its broadest sense, AI is when a computer performs a task that is uniquely human. Here are some of the major milestones:

**1950** –Alan Turing published Computing Machinery and Intelligence. This paper was one of the first attempts to describe “artificial intelligence” as we know it today.

**1996** – IBM supercomputer “Deep Blue” beats world chess champion Garry Kasparov. This worked by searching 6-20 moves ahead at each position and evaluating the best possible option. This was still heavily algorithm based which means the computer wasn’t entirely “learning” by itself. Although this computer could solve a problem as complex as chess, it would still struggle to carry out tasks simplistic to humans. Tasks such as “how many dogs are in this image?” would be out of scope for this type of AI machine. The challenge was not to mimic how humans behave but how they learn.

**2016** – AlphaGo beats Lee Sedol. This was a problem that researchers thought was more than a decade away from being solved. The game of Go is far more complex than chess and this system was able to beat the world number one. The artificial neural network used in this program was trained by playing against both humans and computers.

It was clear from my research that machine learning and deep leaning were the way forward for my AI project. Machine learning is able to mimic how humans learn rather than following hard coded algorithms. For example, when learning a new skill, humans don’t sit with a list of rules to follow; they try something, make mistakes, improve and try again. This is the concept of machine learning. Machine learning models will process large quantities of data and try to decipher it, then it learns from its mistakes and adjusts accordingly. To do this, models were created that had artificial neurons (like those connected by synapses in the brain) connected by weights to make a neural network. These artificial neural networks showed lots of potential and handled complex problems that other algorithms couldn’t. However, even neural networks with 100’s of simply connected neurons could not complete tasks seemingly simplistic for humans. In comparison, the human brain has approximately 86 billion neurons which are intricately connected. Deep learning is intended to bridge the gap between neural networks and human tasks. It includes more neurons, more layers, and more connections between them.

**Computer Vision:**

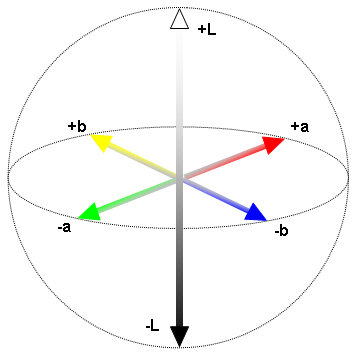
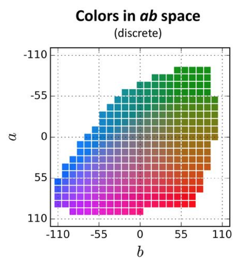
Moving towards the task at hand brought me to computer vision. Not just another buzz word, but a perplexing field, that computer scientists have yet to master even after 60 years of work. The basis of computer vision is the study of visual data; and the aim is to enable computers to see, identify and process images in the same way humans can. When we think of vision as a whole we can split it into two sections: sensing, and interpreting. Although technology has mastered the sensing element of this problem, with modern cameras being comparatively as powerful as the human eye. We have yet to get computers to master the interpretation element of this task, something which even young children can do perfectly. As humans, we use context of everything we’ve ever learned to interpret what our eyes see. For example, if we were asked to identify a dog in an image we would inherently know the features that make up a dog: the shape of the nose, the number and placement of legs etc. But it is a very difficult to incorporate previous knowledge into a computer. This is where deep learning comes into play, if a computer must learn for itself, then it will be able to use context when making decisions.

The possibilities with modern computer vision technologies are endless from industries such as retail, automotive, healthcare, and security. In 2018, Amazon opened its Amazon Go store to the public. With its “Just Walk Out” technology customers need not wait in line to pay for their purchases. This has the potential to change the retail world drastically. The automotive industry can also benefit from these technologies. With the World Health Organisation stating that more than 1.25 million people die each year due to traffic incidents it is vital that we utilize new technologies to provide safer road experiences. According to the WHO research, the majority of these road incidents are caused by human error and inattention. Currently, Tesla and Google are leading the way in the world of autonomous driving. These “self driving” cars have sensors that can detect movement of pedestrians, cyclists, vehicles and road works from up to three football fields away. Furthermore, advances in computer vision could literally be saving lives as Gauss Surgical has developed blood monitoring solutions which are accessible on an iPad app! The Triton OR app which captures images of blood has been reported to recognize hemorrhage better than the human eye.

The branch of computer vision that I was delving into was colourisation. Colourisation is not a new concept, in 1902 Marie-Georges-Jean Méliès, a French illusionist created “A Trip to the Moon” which was one of the first works to be hand coloured. In 2004, the movie Mughal-e-Azam, a blockbuster movie released in India in 1960 was remastered in colour and became a huge hit for the second time. For many years this was a very tedious and time consuming task, but nowadays the process is becoming more automated with the help of deep learning techniques. At ECCV (the European Conference on Computer Vision) 2016, Richard Zhang, Phillip Isola, and Alexei A. Efros published a paper titled Colorful Image Colorization with some very impressive results. The creators of this paper presented a Convolutional Neural Network (CNN) for colourising greyscale images. They trained the network with 1.3M images from ImageNet training set. At a glance, this may seem to be an easy task as humans are capable of making reasonable guesses of what most colours in an image would be; but it’ still impossible to know for certain what colour a t-shirt would be from its greyscale representation. The main aim of this paper was to produce plausible suggestions of colourisation which could fool a human observer.

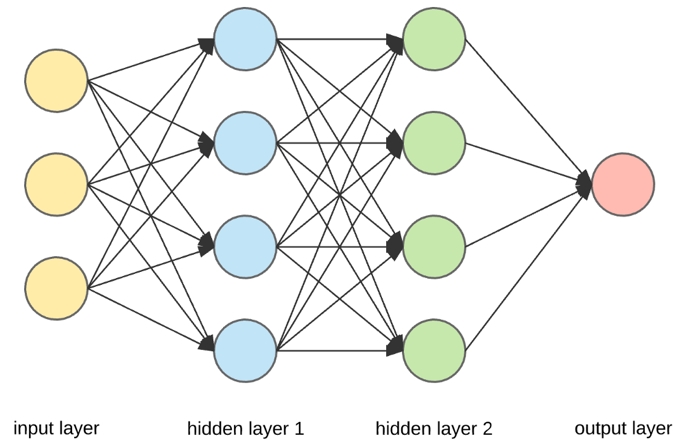
**The Colourisation Problem:**

Our programme will work with the CIE Lab colour space. This is a 3 –channel colour space which is encoded by the **a** (green-red), **b** (blue-yellow) channels and the **L** (lightness) channel. (fig. 1) The L channel can be thought of as the “intensity” of our image, which accounts for a large percentage of what humans see in an image. With greyscale images, every pixel can have a value 0-255 (black –white). We can think of our greyscale input image as our L-channel. For ease of use, the ab space is quantized into 313 bins. (fig. 2) Now we have our L channel which takes values from 0-255, we need to find the ab value from 0-312 for each grey pixel.

Fig. 1 Fig. 2

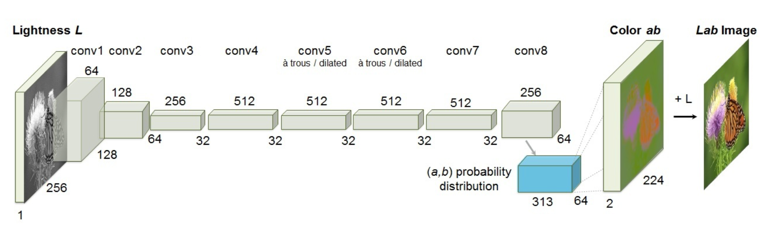
**CNN Architecture:**

In its simplest terms, a neural network can be thought of as a black box with an input -> some computation -> an output. From figure 3 we can see that the input is connected to a hidden layer which can be connected to many more hidden layers before giving an output. The nodes are all connected by weights and these weights determine how much impact each node will have on the next. Between layers we have activation functions which essentially define if a node should be “activated” or not based on the weighted sum. This decides whether the information the neuron is receiving is relevant or should be ignored. The output of the activation function is sent to the next layer of neurons as input.

 Fig. 3

All neural networks are trained using a loss function. This loss function gives an indication of how accurate our prediction is. The end goal is to minimize the loss to ensure our predictions are correct. The loss function is calculated by comparing our mapped prediction with the mapped truth and analyzing the difference. We want to use our loss function to adjust the weights in the model and steer them towards the correct ground truth. This is done through a process called backpropagation. Backpropagation is about determining how changing each weight will impact the overall loss of the network. This happens by starting at the end of the network and working backwards calculating how much each weight is calculating to the overall error/loss. The values found to cause the highest error will be changed more than those that contribute less to the error. This is the “improvement” step – the program is learning from its mistakes. The final weights of the fully trained model are used for our ultimate predictions.

**The Colourisation Model:**

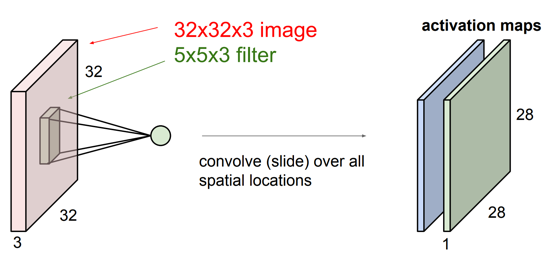
Fig. 4

The model proposed in this paper, shown Fig. 4, was a VGG style network with 2 or 3 convolutional layers followed by a Rectified Linear Unit (ReLU) function. This ReLU function is our activation function and is one of the most common in use, especially for computer vision problems. Neural networks use activation functions to allow the network to solve more complex problems. The benefit of using the ReLU function over other activation functions is that it doesn’t activate all neurons at the same time. Any negative input will be converted to zero and the neuron will not be activated. This means that only a few neurons are activated at any given time making it efficient and easy for computation.

If we take our greyscale image as input X, and feed it through the network G, our result will be ẑ = G(X). This output will give us a height a weight and a vector Q (=313). Each value in the vector Q will represent the probability of the pixel belonging to that class in the ab colour space (Fig. 2). Our goal is to find a single pair of ab channel values for each pixel. We cannot simply take an average of all the probable values as it will result in a mucky, unnatural colour. For example, an apple could be green but it could also be red. If we take the mean of these two colours we would get an unappetizing, brownish colour. Another solution would be to take the mode value and have a result of either green or red. While this gave vibrant colours, it sometimes broke spatial consistency. The creator’s solution was to find somewhere in the middle of these values. This resulted in the annealed-mean, which was the final value of our corresponding ab pair. Lastly, our image is upsampled back to its original size and the ab and L channels are added to produce the final colour image.

**The Convolutional Step:**

Above, I’ve described the system as a whole but now I will jump into the details of what happens at each layer. As previously mentioned, this model consists of two or three convolutional layers followed by the ReLU function, but what is actually happening at each layer? In mathematics, convolution is a mathematical operation on two functions. That is essentially what is happening at our convolutional layers. We slide a filter over our image step by step and calculate a dot product at each location. (Fig. 5) The filters will always have the same depth as our image. In our colourisation problem, the training data consists of a vast amount of colour images and their greyscale versions. The objective of the neural network is to find characteristics that link these greyscale images with their coloured versions. Firstly, it will look for simple patterns: lines, dots, curves. When scanned through again, the system notices small patterns it has seen before. To gain a higher understanding, we decrease the size of the image so the filter is covering a greater area. Combining these filters allows the system to identify more significant patterns such as the shape of an eye or the mouth. For colouring networks it is important not to distort the image as we want the pixel location to stay the same. To ensure this, we avoid pooling layers and instead use a stride > 1 to decrease the width and height at each step.

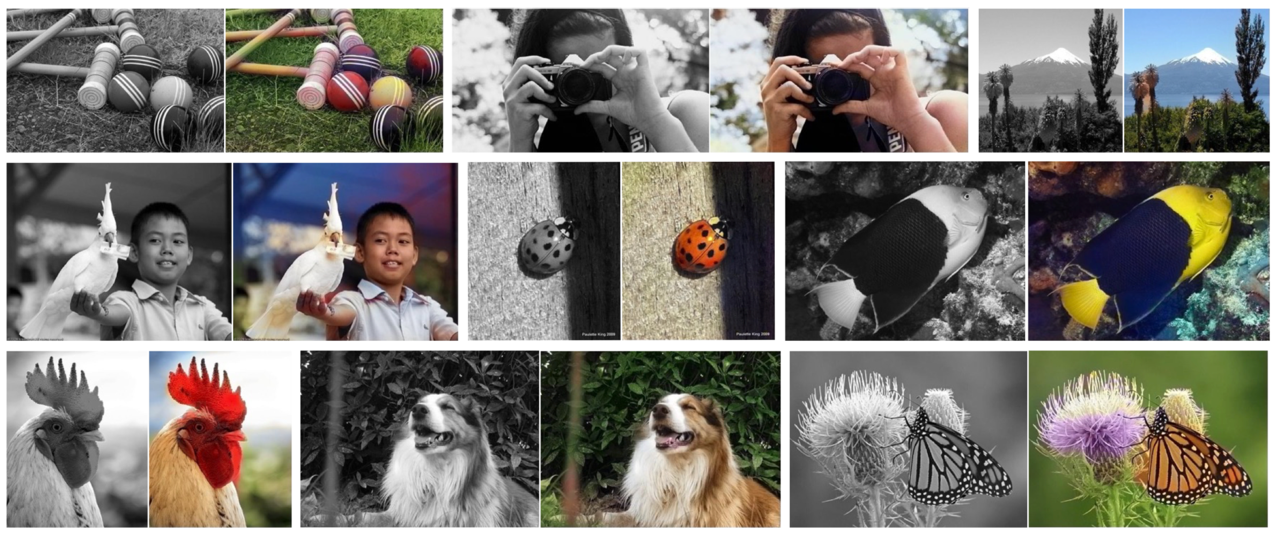
Fig.5

It is a combination of these activation maps that gives the system context to understand the image.

**The results: (Fig 6.)**

The results from this paper were immensely impressive and were shown to successfully fool humans in 32% of the trials. This beats previous methods by vast amounts. Furthermore, this approach results in state of the art performance on feature learning benchmarks. More results can be found here: <https://arxiv.org/pdf/1603.08511.pdf> . These result were adapted by Sunita Nayak, to use the pre trained model for videos as well as images. More information: <https://github.com/spmallick/learnopencv/tree/master/Colorization>. I was even able to test out this technology on several of my own images and videos which can be seen on my corresponding presentation.

Overall, I believe the process of colourisation can be as much of a scientific one as it is artistic and these advances in technology could open exciting doors in fields such as cinematography, and historical research. The experimentation possibilities with this system are endless as users could retrain the model with a different data set, adjust the parameters and observe new results.

 Fig. 6

This was a very exciting and interesting first machine learning project and has just fed my curiosity to explore the field. I will continue my research beyond this internship and hopefully, will soon be able to design my own models. I hope that my journey can encourage others to dip their toes into machine learning as it is truly fascinating work which opens a world of possibilities.