Lecture 1

Gidon Rosalki

2025-10-19

1 Introduction

Given a scene, a camera turns it into pictures / video, which may then undergo computation resulting in better pictures. We may also gain information from the scene through computer vision, perform video sound analysis, and use them for self driving AI.

There are 2 main areas to computer science. There is classical CS, where we invent algorithms, and consider their complexity, algorithms, computer graphics, databases, architectures, cryptography, languages and compilers, and networking. This overlaps in "cyber" with the field of learning from data, and AI, which includes computer vision (this course), speech analysis, natural language, deep learning. Here stage 1 is preprocessing and tokens, and stage 2 is the deep learning. An example of AI classification is facial recognition. This will not be discussed in this course.

We state that in nature, vision = intelligence = movement. Only intelligent organisms that move can see. Bacteria and plants do not see. Visual recognition begins at an early age, with babies being able to track their mothers. Most of the of the human brain is involved in visual processing. We see great variety in this in the real world. Most predators have eyes directed forwards, improving depth perception, and aiding their ability to hunt. Most prey have to be able to escape predators from any direction, and so have more sideways directed eyes, to improve their ability to see predators in their peripheral vision.

Image processing has the application of image enhancement, such as taking a photo from a hazy day, and removing the haze from the picture. Additionally, to can be used to increase the dynamic range of an image, allowing us to see the details that would otherwise be obscured in darkness.

An example of a use that was a previous exercise for this course was taking a series of pictures that together form a panorama, and turning it into the 3D image.

1.1 Computer vision

Computer vision started around 1964, but nothing worked for 50 years. Around 2014, the introduction of neural networks allowed the beginning of more effective research. Computer vision products are now commonly used:

- Face detection is included in every camera
- Face recognition in Facebook, Google, Passport control
- Autonomous driving
- Medical diagnoses
- OCR
- Image and video editing and generation

1.2 Grade

3 exams throughout the semester, of 1 hour, containing 2 questions. The final average is normalised to 83%-85%. Of the 6 questions across these exams, the best 5 are taken. There are 5 individual programming homeworks. The final grade is comprised of 25% for each exam, 23% from the homeworks, and 2% attendance.

1.3 Image formation

1.3.1 Luminance

Light is emitted by light sources, and reflected from objects. The reflected light is sensed by an eye, or camera. We may calculate the intensity by multiplying together the reflectance by the emitted light:

$$I = L \cdot r$$

1.3.2 Reflectance

We have 2 main types of reflectance, shining (specular) and matt (diffuse). Under specular reflection, a ray is reflected from the surface with the same angle to the normal. Angle of incidence is equal to angle of reflection. However, on a matt surface, given a light ray incident at an angle, all angles will have the same chance of reflection.

1.4 The eye

Light passes through the iris, until it hits the retina at the back of the eye (different from a typical image sensor, since it is on a ball). The retina is formed of around 10^8 rods, which can see black and white, 10^7 cones for RGB colour, and 10^4 nerves. There is therefore about a $1 : 10^4$ reduction. This reduction takes place since if we have 10^8 nerves connecting to the eye, we would need a metre of space for all the nerves to fit. Rods are very sensitive, even in low light, whereas cones are more for bright light. We can only see a small amount of the electromagnetic spectrum, from around $350nm - 780nm$. Within there, the blue cones fire at around 420nm (high frequency), the rods at 498nm, green at 534nm, and finally red at 564nm (low frequency).

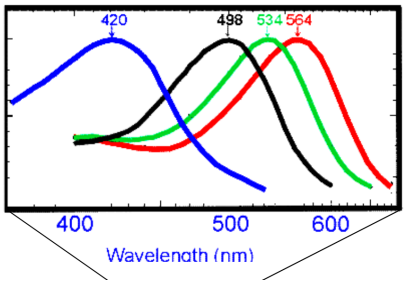

Figure 1: Visible frequencies

1.5 Image digitisation

There are 3 stages to image digitisation:

1. Transforming the 3D world into a 2D image. This involves perspective projection (optics, continuous)
2. Sampling the image plane with a finite number of pixels
3. Quantising the colour / grey level with a finite number of colours (such as 8 bits per colour)

The simplest form of camera is a pinhole camera, or camera obscura (latin for dark room). If we place an image plane in front of an object, no image is generated. To create an image, each image location should get a ray from a **single** point in the scene. This is done by blocking all the rays except one by placing a wall with a very small hole in it in front of the image plane. Since very little light passes through, we need to place the image plane in a dark room, such that the low level of light passing through the pinhole is sufficient to be visible.

1.5.1 Perspective projection

When transforming the 3D world into a 2D image, we use continuous perspective projection, where all rays pass through one focal point, and where they land on the image plane is where the point is stored.

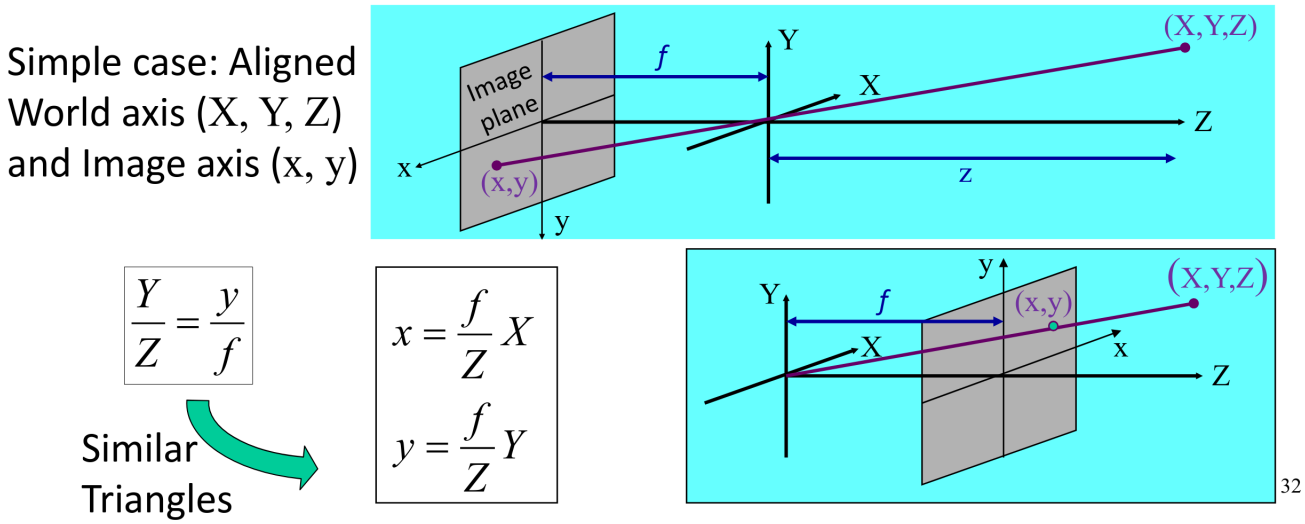

Figure 2: Perspective projection (f is the focal length)

In order to transform the world to camera we have the following equations:

$$x = \frac{f}{Z}X$$
$$y = \frac{f}{Z}Y$$

Where X is parallel to x, Y to y, and both use the same units. In the general case, there is a transformation matrix between world axis and camera axis.

1.6 Cameras

In 1975, Bruce Bayer suggested the colour filter array used in most digital camera. $\frac{1}{4}$ pixels detect red; $\frac{1}{4}$ pixels blue; $\frac{1}{2}$ pixels green. Other colours are invented through demosaicing, averages, and other proprietary methods.

There are a few different colour spaces. In typical camera / projector we use an additive colour space of RGB. We add together different amounts of red, green, and blue to create new colours. However, printers use the CMYK colour space, since they print with ink which *absorbs* colours instead, and is thus a subtractive colour space. Mixing these colours together will subtract out colours, resulting in RGB and black. However, to save on ink, most printers include a special black ink cartridge (K).

The first TVs were in black and white. When transferring to colour TV, since all the TVs at the time were in black and white, they created a transmission standard which could be understood by both colour TVs, and black and white.

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Where Y is the luminance. In Israel, it was decided that colour TV was wrong, and so would noise the channels of I and Q. Immediately, enterprising people created denoisers, which everyone promptly bought to be able to see colour TV.

1.7 Point operations

Given an image, we can generate new pixel values $g(x, y)$ based on the input pixel value $f(x, y)$. For example,

g $(x, y) = 255 - f(x, y)$ This takes the negative of an image. This operation depends only on the value of the pixel, and not surrounding pixels, or the location. In general:

1.7.1 Gamma correction

Gamma correction is used to overcome non linear responses of cameras, displays, and eyes.

$$T(u) = \text{Max} \cdot \left(\frac{u}{\text{Max}}\right)^{\gamma}$$

This is also achieved through lookup tables.

1.7.2 Common histogram

Images can give us histograms with the distribution of different colours in the picture.

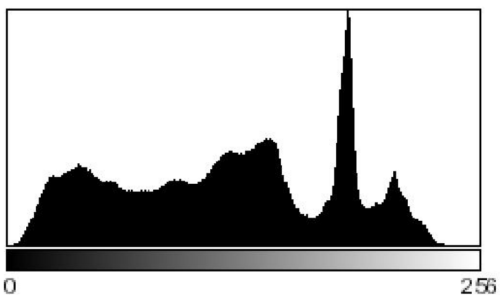

Figure 3: Common histogram

The histogram is invariant to pixel locations. Different pictures can give the same histogram. Consider how many pictures there are that have all the colour split into two discrete locations in the graph. Typically, the histograms will have very small amounts at either end, with most of the frequencies in the middle. When there is nothing in the higher, and lots in the lower, we know the image is probably under exposed, and if there is lots in the upper frequencies, it is probably over exposed.

Consider what JPEG does to the histogram:

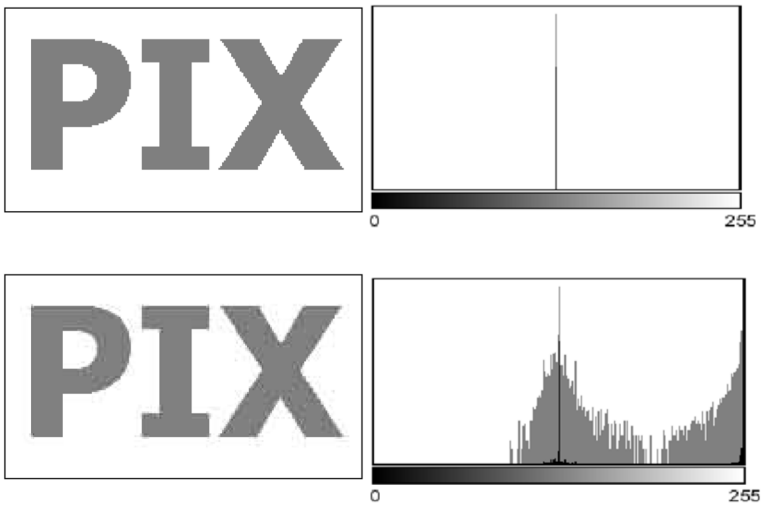

Figure 4: JPEG effects

It has created a lot of new colours, in trying to condense down the lines of the image.

These histograms can be useful for cut detection in videos. Similar images will have similar histograms, especially when the camera is recording at 24fps, the scene cannot change that much from frame to frame. However, the histogram will change significantly with a cut, so computing the distance between the colour histogram of successive frames can detect these cuts.

The vector distance between histograms is simply the absolute sum of the difference between the buckets. However, two very different sets of histograms can have the same vector distance, even if one set is much closer to its pair than the other. In this case we may use the distance between **cumulative** histograms. So, if we take a histogram where all the pixels are 1, one where they are all 2, and one where they are all 5, the cumulative distance will show that the first two are much closer together, then either of them are to the last.

We may use histogram equalisation, which expands the histogram to the entire space fo the image to gain a lot more contrast in a given image:

Figure 5: Histogram normalisation

In order to compute the histogram equalisation, we compute the cumulative histogram $S(i)$ from the histogram $h(i)$, and change every original grey level i to $S(i)$. Next we perform a linear *stretch* the new grey levels back to $[0, \ldots, k]$. Given m the lowest grey level in the input image, and q the highest grey level, we map $S(m) \to 0$ and $S(q) \to K$. We wind up with the computation:

$$i \implies K \frac{S(i) - S(m)}{S(q) - S(m)}$$