# Part 1 - Written Part

## Solution to Problem 1.

1. Bit rate=Sampling(samples/second) \*Quantization (bits/sample).

Hence we know that the width and height of the image is 450 and 520 and the frame rate is 25 Hz. So according to the above formula = Sampling=520\*450\*25 samples/second.

Quantization can be found according to the 4:2:0 format for the Y, Cr, Cb as the average bits per pixel or sample= (4\*8+1\*8+1\*8)/4= 12 bits per sample.

Hence Bit rate= 520\*450\*25 \*12=70.2 Mbps.

1. If we re quantize the Y, Cr, Cb 4:2:0 format with only 6 bits for the chrominance channel then average bits per pixel will change according to the formula:(4\*8+1\*6+1\*6)/4=11 bits per pixel.

Hence the Bit rate would change to =520\*450\*25 \*11=64.35 Mbps.

Hence if this made to run for 10 mins (600 seconds) = Disk capacity=600\*64.35 megabits=4.49 gigabyte. Hence the storage on the disk=4.49 Gigabytes.

## Solution to Problem 2.

1. Since it is clearly mentioned in the question that 32 levels are present, we can definitely conclude that number of bits=log2(32) = 5 bits per sample. Since there are 32 levels or samples, we will require 32\*5= 160 bits in total.
2. Quantized Sequence (If we consider the signal values only):

1.75, 2.25, 2.25, 3.25 3.25, 3.25, 2.5, 2.75, 2.75, 2.75 ,1.5, 1.0, 1.25, 1.25, 1.75, 2.25, 2.25, 2.25, 2, 2.25, 1.25, 0.25, -1.25, -1.25, -1.75, -1, -2.25, -1.5, -1.5, -0.75, 0, 1.

Quantized Sequence (If we consider the level or bucket values only):

22, 24, 24, 28, 28, 28, 25, 26, 26, 26, 21, 19, 20, 20, 22, 23, 24, 24, 23, 24, 20, 16, 10, 10, 8, 11, 6, 9, 9, 12, 15, 19.

This follows since the range is equal to 8 and we have 32 levels, height of each level=8/32=0.25. hence this relates to 5 bits per level and since the level 0 is placed at -3.75 instead of -4 we have the following level or bucket values.

## Solution to Problem 3.

1. Speed of the car = 36 km/hr.

Diameter (d)=0.4244 m.

Distance covered in rotation =circumference of the wheel= pi\*d

=3.142\*0.4244

=1.333m

Conversion of speed to meters/second=36\*5/18=10 m/s.

Hence the rotations per second= 10/1.333=7.5 rotations/second.

Since we are sampling at a rate of 24 frames per second which is greater than twice the Maximum frequency (2\*7.5) = 15 rotations per second. The Nyquist criterion is satisfied and the rotations seen in the movie would be at 7.5 rotations/second.

1. Since the film rate in this case is 8 fps which is lesser than the required Nyquist rate of 15, there is Aliasing effect that occurs and since such an effect occurs, a copy of the frequency at 7.5 is created at 0.5 rotations/second. The aliased frequency is given by the formula

=|Film rate\*n-rps|

Where n is the closest integer value to the film rate and the rotations/ second. Hence in the above case:

Film rate=8,

Rotations/second(rps)=7.5

Hence the closest integer value for both of them or the (Greatest Common Divisor) = (8/7.5) =1.

Hence Aliased frequency=|8\*1-7.5|=0.5 rotations/second.

# Part 4 –Analysis Questions on video conversion

SEAM Carving.

For the Conversion to HD2SD and SD2HD

In this process Seam carving was used and the outputs are display below with lowest stretching effects are seen and even maintain the aspect ratio. Since this technique is content aware image resizing, blurring effects are nullified. Seam carving also allows manually defining areas in which pixels may not be modified, and features the ability to remove whole objects from photographs.

It functions by establishing a number of seams (paths of least importance) in an image and automatically removes seams to reduce image size or inserts seams to extend it. The purpose of the algorithm is image retargeting, which is the problem of displaying images without distortion on media of various sizes using document standards.

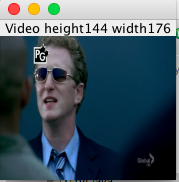


Fig 1: HD2SD conversion by content aware Seam carving.



Fig 2: SD2HD conversion by content aware Seam carving.

Since this process can be used to stretch and elongate pixels to any size without any disturbance and blurring while maintaining the aspect ratio, it was chosen.

Different ways to carry out such a process of maintaining the aspect ratio while not introducing blurring while conversion from SD2HD or HD2SD include:

1. Cropping content in the majority of image to a pixel ratio of 4:3 while wanting to achieve HD2SD and using interpolation techniques and using 16:9 ratios while converting SD2HD. It is important to warp the edges while doing this process in order to reduce the blurring.

2. Padding the image to the actual ratio by increasing or decreasing the size commonly known as Black boxing or padding. This causes the conversion from an exact 4:3 to 16:9 and vice versa.

Both the above cases have some amount of losses and blurring to a small extent. Post and pre filtering can be used to reduce the blurring. The examples below show the effect of reduction in blur using bilinear in comparison to a nearest neighbor.

**Reduction with Post and Pre-Filtering using Bilinear instead of Nearest Neighbor.**

Since the Nearest Neighbor interpolation that we applied to interpolate appeared blocky, hence different interpolation techniques such as bilinear interpolation, bi-cubic interpolation need to be carried out.

Given below are the different Input images and the Bilinear Interpolated versions of the same. It can be clearly visible from the below images the difference between the nearest neighbor interpolation and bilinear interpolated versions. It is to be noted that while anti-aliasing can be done by filtering. The 3x3 mean or averaging filter produces a better result as compared to the 5x5 mean filter because more blurring is obtained in the latter version. It is important to note that Pre filtering process is carried out in case of hd2sd conversion and post filtering is carried out for sd2hd for anti-aliasing purposes.

It can be clearly visible that bilinear interpolation is far better than nearest neighbor.

Case 1:

Original Image. SD image.



Fig 1:SD image.

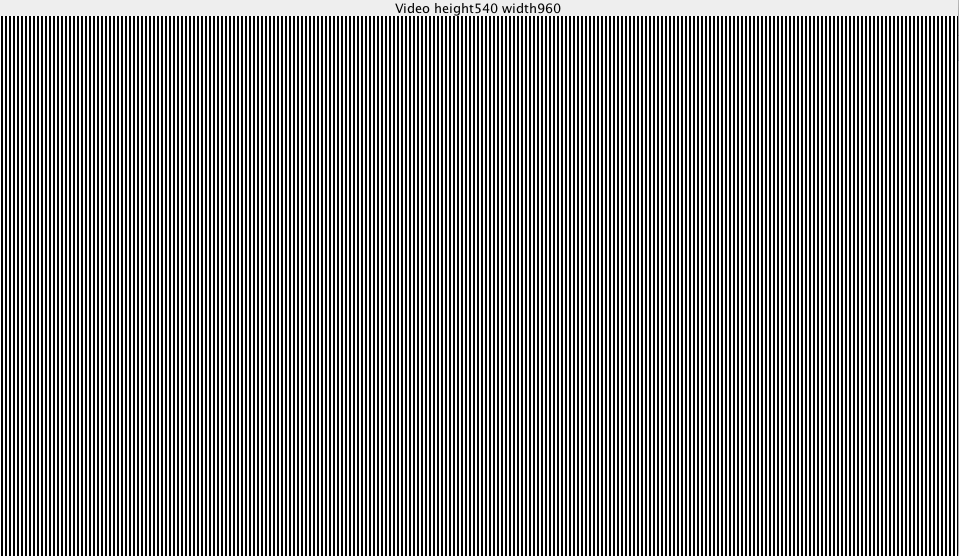


Fig2: HD image.

HD2SD conversion by Nearest neighbor and Bilinear Interpolation.

With Aliasing.

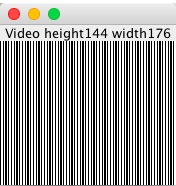


Fig 3: HD2SD by nearest neighbor (right) and by Bilinear interpolation (left).

HD2SD conversion by Nearest neighbor and Bilinear Interpolation.

Without Aliasing.

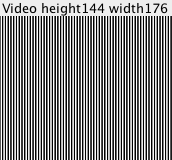
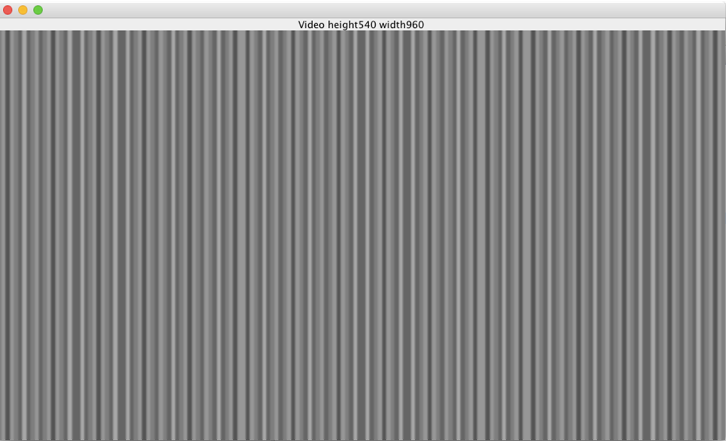


Fig 4: HD2SD by nearest neighbor (left) and by Bilinear interpolation (right).

SD2HD conversion by Nearest neighbor and Bilinear Interpolation.

With Aliasing



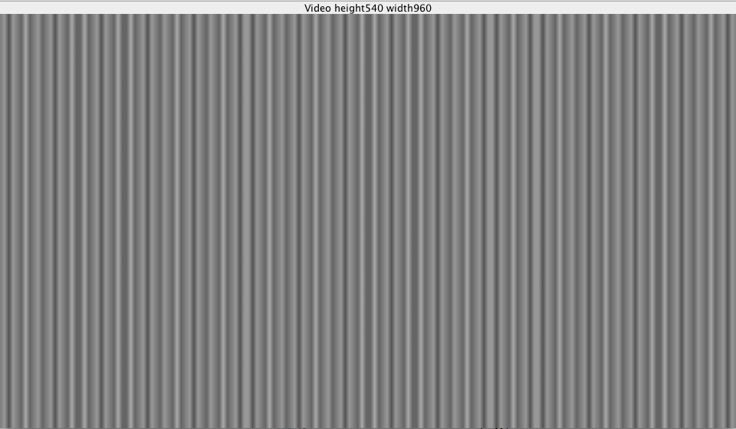
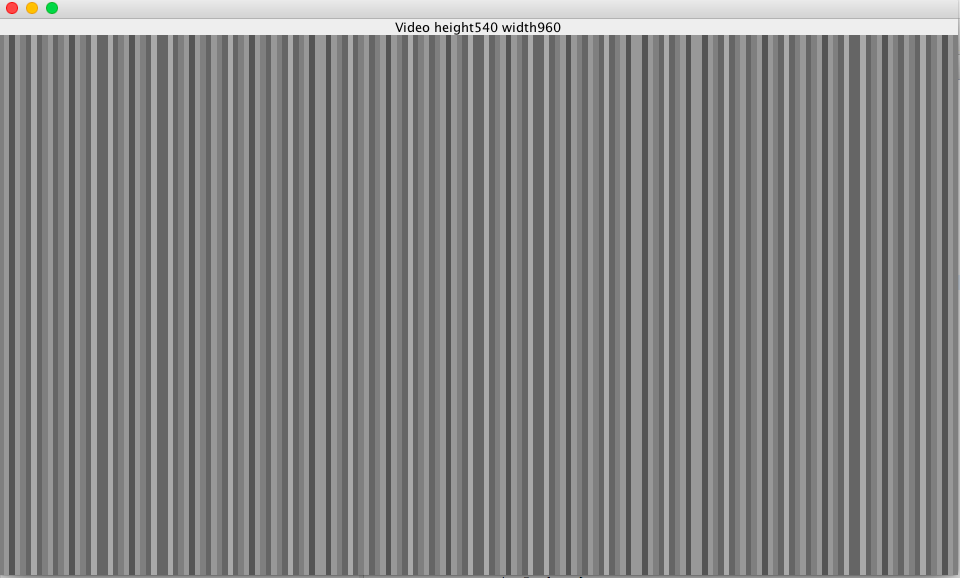


Fig 5: HD2SD by nearest neighbor (up) and by Bilinear interpolation (down)

SD2HD conversion by Nearest neighbor and Bilinear Interpolation.

Without Aliasing



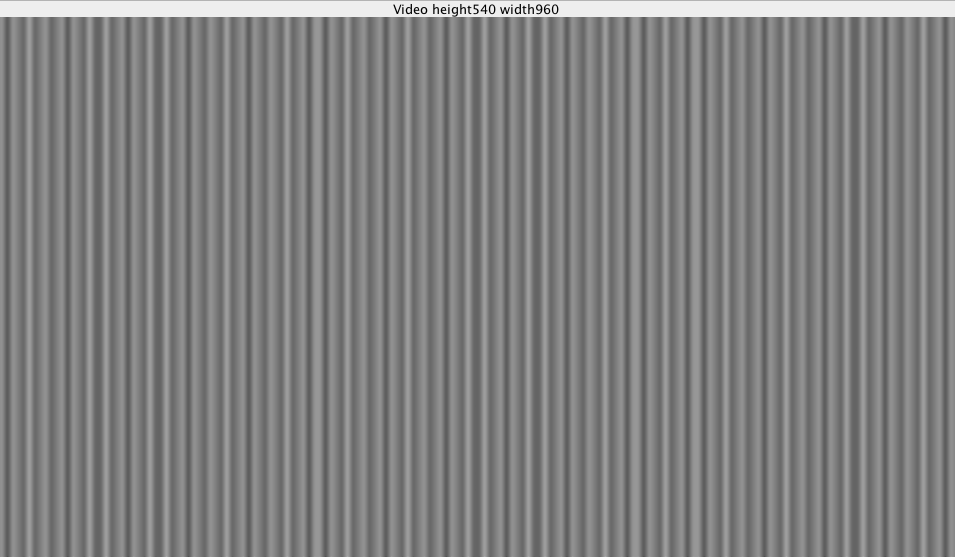


Fig 6: HD2SD by nearest neighbor (up) and by Bilinear interpolation (down) without aliasing

Case 2:

Prison image.

Original SD image.

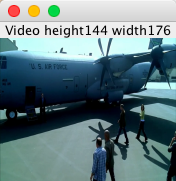
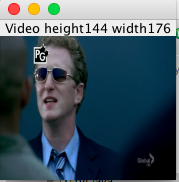
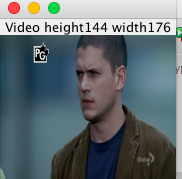


Fig 7: Original SD prison image.

Original HD image.

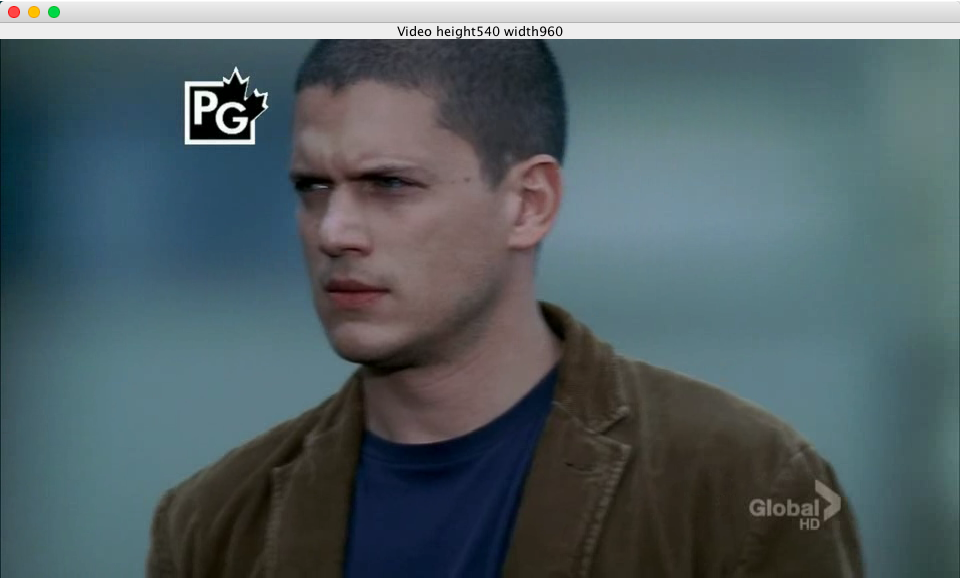


Fig 8: Original HD prison image.

HD2SD conversion by Nearest neighbor and Bilinear Interpolation.

With Aliasing.

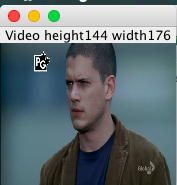
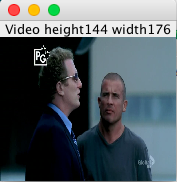
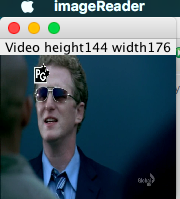
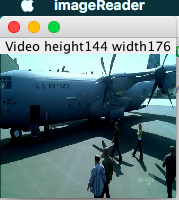


Fig 9. HD2SD conversion using nearest neighbor. Blocky patches seen on the wing of the flight and faces which is undesirable.

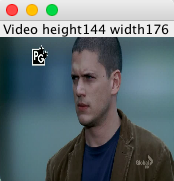
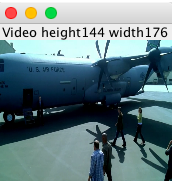


Fig 10. HD2SD conversion using bilinear interpolation. Better display than the nearest neighbor but still patchy in nature due to aliasing effect which can be removed by filtering.

HD2SD conversion by Nearest neighbor and Bilinear Interpolation.

Without Aliasing.

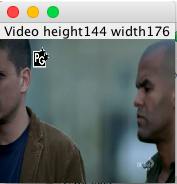


Fig 11. HD2SD conversion using nearest neighbor. Filtering done by 3x3 mean filter for nearest neighbor.

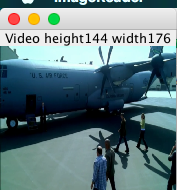
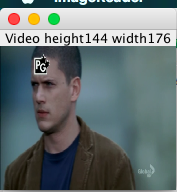
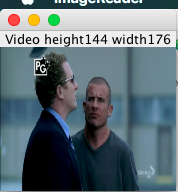


Fig 12. HD2SD conversion using Bilinear interpolation. Filtering done by 3x3 mean filter, better output as compared to the nearest neighbors if the wings and the blades of the fan are considered.

SD2HD conversion by Nearest neighbor and Bilinear Interpolation.

With Aliasing.



Fig 13. SD2HD conversion using Nearest Neighbor. No post filtering is done and the patchy image is clearly visible with stretching effects.



Fig 14. SD2HD conversion using Bilinear filtering. No post filtering is done and better results as compared to nearest neighbor are obtained. Still post filtering is needed for anti-aliasing.

SD2HD conversion by Nearest neighbor and Bilinear Interpolation.

Without Aliasing.





Fig 15. SD2HD conversion using Nearest neighbor. Post filtering is done and better results as compared to nearest neighbor with aliasing effects. Still stretching effects observed.



Fig 16. SD2HD conversion using Bilinear interpolation. Post filtering is done and better results as compared to nearest neighbor. Stretching effects observed are reduced considerably.