سوال اول

a) RGB to CMYK
K = 255 – max(R, G, B) = 255 – 255 = 0
C = 255 – K – R = 255 – 0 = 255
M = 255 – K – G = 255 – 255 = 0
Y = 255 – K – B = 255 – 100 = 155

b) CMY to RGB
R = max – C = 255 – 80 = 175
G = max – M = 255 – 43 = 212
B = max – Y = 255 – 100 = 155

c) CMYK to RGB
R = 255 × (1 - Cyan ÷ 255) × (1 - Black ÷ 255) = 255 × (1 - 115 ÷ 255) × (1 - 155 ÷ 255) = 55
G = 255 × (1 - Magenta ÷ 255) × (1 - Black ÷ 255) = 255 × (1 - 87 ÷ 255) × (1 - 155 ÷ 255) = 66
B = 255 × (1 - Yellow ÷ 255) × (1 - Black ÷ 255) = 255 × (1 - 0 ÷ 255) × (1 - 155 ÷ 255) = 100

سوال دوم

The correlation between two signals or simply the cross correlation is a standard tool for evaluating the degree to which two signals are similar. Normalized cross correlation has been commonly used as a metric to evaluate the degree of similarity (or dissimilarity) between two compared images. In seismology, correlation is often used to search for similar signals that are repeated in a time series – this is known as matched filtering. Because the correlation of two high amplitude signals will tend to give big numbers, <u>one cannot determine the similarity of two signals just by comparing the amplitude of their cross correlation</u>. If the two image regions are feeble variations of the same scene, then the heuristic change has little effect. On the other hand, if there is a large difference in contrast between the matched regions, then the modified normalization potencies the correlation to zero (i.e. effectively uncorrelated). Finally, contrast normalization allows matching with regions that are principally uniform except for random or texture variations.
By my understanding, this algorithm is used when we have a small image and a larger image, and we want to figure out which subset of the large image the smaller one is and we want to do it by finding the correct coordinates (Like trying to find a specific string in a text).The steps are as follows:
1- Read images
2- Choose Subregions of Each Image
3- Performing Normalized Cross-Correlation and Finding Coordinates of Peak.
4- Find the Total Offset Between the Images
5- Checking whether the small image Was Extracted from the large image.
6- Padding the small Image to the Size of the large Image.
7- Transparently Overlay small Image on large Image.

Sources:
https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.675.1379&rep=rep1&type=pdf#:~:text=The%20correlation%20between%20two%20signals,which%20two%20signals%20are%20similar.&text=Normalized%20cross%20correlation%20(NCC)%20has,dissimilarity)%20between%20two%20compared%20images.

https://www.youtube.com/watch?v=ngEC3sXeUb4

سوال سوم

The Hessian affine detector algorithm is almost identical to the Harris affine region detector. In fact, both algorithms were derived by Krystian Mikolajczyk and Cordelia Schmid.

The Hessian corner direction algorithm is implemented based on Hessian matrix (H) computation. Note that Hessian matrix is defined for a given image I at a pixel p is
H1(p) = |Ixx(p) Ixy(p)| |Ixy(p) Iyy(p)|
such that eigen-decomposition (spectral decomposition) of this matrix yields two eigenvalues as: Lambda1 and Lambda2. If both Lambda1, Lambda2 are large, we are at a corner.

In Harris detection, rather than considering the Hessian of the original image I (i.e. second-order derivatives), we use the first-order derivatives of the smoothed version for some Gaussian filter with standard deviation sigma > 0. We then calculate the corner strength using two different functions and then compare their efficiency.

Harris output: A very precise corner detector.

Hessian output: Responses mainly on corners and strongly textured areas, because the second derivative is more sensitive to change.

سوال چهارم

Used this link to calculate the Eigen Values:
https://www.symbolab.com/solver/matrix-eigenvalues-calculator

M1 = [94.44, 50.87]
M2 = [163, 0]
M3 = [0, 164]
M4 = [0, 0.5]

M1’s eigen values are both large, so it’s a corner. M2 and M3 are edges and M4 is a flat surface. Between M2 and M3, M2 has a larger Ix, so it means it had more changes in the x axis so it’s a horizontal edge, and with the same reasoning, M3 is a vertical edge. Therefore, the final answer is as follows:
M1: Green, M2: Blue, M3: Red, M4: Yellow