Question-1:

Such an architecture can not perfectly classify all of the training examples. As we know, Convolution layers are linear. therefore, any layer composition made out of linear layer is still linear. Assume our inputs are 16 dimensional binary vectors, and we wish do discriminate between two patterns A and B, that may be arranged in any conceivable translation. if we change the pattern to the right whatever falls on the right side will fall on the left. As a result, our class has 16 cases of A and 16 examples of B that our classifier need to distinguish. if our classifier is to correctly identify all 16 cases of A, it must also classify the average of all 16 occurances as A. Because four out of sixteen values are

active, the average of all instances is the vectors (0.25, 0.25, ..., 0.25). similarly for it to correctly classify all 16 instance of B it must also classify their average of B. since this vector can not possibly be classified as both A and B. this dataset must not be linearly separable. in general, no linear classifier can be expected to discover a pattern in all potential translation. This is a significant drawback of using classifier as the foundation for a vision system. Therefore, above mentioned architecture can not perfectly classify all of the training examples.