1 True or False, 5p

a) False, the median filter is a non-linear operation. Only linear filters can be (fully) understood using the Fourier transform.

b) True, morphological erosion is a translation invariant operator.

c) True, coding methods like Huffman coding can be applied on basically any type of data. Since it is non-lossy.

d) False, additional features does not help if you have a limited supply of training data.

e) False, but the third moment of the object boundary will measure skewness.

f) True, the sum of all "bars" in the image histogram is equal to the number of pixels in the image for an un-normalized histogram.

g) False, brightness and contrast adjustment are in general irreversible. I.e. multiply with 0...

h) False, you can easily construct a non-constant neighborhood where the coefficients of a Laplace filter kernel sums to 0.

i) False, pincussion and barrel distortion are geometric distortions. They are not shift-invariant.

j) True, pseudo coloring is used to visualize differences using colors instead of gray-scale.

2 Morphology 5p

a) Look up erosion in lecture notes. Does the structuring element fit the set? Find all places where the structuring element is fully contained in the image set X. The set of the origins of those structuring elements is the erosion result.

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0	0
0	0	0	0	0	1	1	0	0
0	0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

b) Look up dilation in lecture notes. Does the structuring element hit the set? Find all places where the *reflection* of the structuring element touches the image set X. The set of the origins of those structuring elements is the dilation result. (this structuring element is symmetrical)

0	1	0	0	0	0	1	0	0
1	1	1	0	1	1	1	1	0
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1
0	0	0	1	0	1	0	1	0

c) Look up closing. Dilation followed by erosion.

0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	0	0
0	1	1	0	1	1	1	1	0
0	1	1	1	1	1	1	0	0
0	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	0	0
0	0	0	1	1	1	1	0	0
0	0	0	1	0	1	0	1	0
0	0	0	0	0	0	0	0	0

d) Loop up opening. Erosion followed by dilation.

0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0
0	0	0	0	0	1	1	1	0
0	0	0	0	1	1	1	0	0
0	0	0	0	1	1	1	1	0
0	0	0	0	1	1	1	0	0
0	0	0	0	1	1	1	0	0
0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0

e) Design a SE B1 for the object foreground. Design a SE B2 for the object background. E.g.

$B1 =$

0	0	0	0
0	1	0	0
0	1	0	0
0	1	1	0
0	1	1	0
0	0	0	0

$B2 =$

1	1	1	0
1	0	1	0
1	0	1	1
1	0	0	1
1	0	0	1
1	1	1	1

3 Image Restoration, 5p

a) Many things needs to be done. First you have to find estimates of H (the spectrum of the point spread function), N (noise) and F (the uncorrupted image). These can all be hard to estimate. Secondly you need to fourier transform your signal and possibly also add padding to the borders, to avoid undesired edge effects. The noise could be estimated from a part of the image where you know the signal, e.g. a constant area. The point spread function needs to be estimated from a bright spot or a pair of orthogonal lines.

b) The Wiener filter is optimal in a least square sense.

c) h is the point spread function. The easiest way to estimate it is to look at a small bright spot and see how it is transformed.

d) N/F is usually simplified by a constant factor K, which avoids the difficulties involved with estimation of N and F.

4 Coding and compression, 5p

a) $f(x_i) = x_i$, if $i = 0$ and $f(x_i) = x_i - x_{i-1}$ if $i > 0$. An image needs to be "serialized" by enumerating the pixels in some continous order, e.g. a zig-zag-pattern.

b) 8, -1, -1 -1, 0, 0, -1, 0, -1, -1, -1, -1, 2, 0, -1, -1

c) The histogram is rather flat (no super-high peaks) and it will not be possible to compress easily using Huffman encoding. Its entropy is high.

d) The histogram is this, encoded as (symbol, count): (-1, 10), (0, 4), (2, 1), (8, 1) and it has a high peak for -1. Thus it is likely that it has a better compression ratio. One could check this further by e.g. computing the entropy.

5 Fourier domain, 5p

a) Fill in $k = 0$. We see $e^0 = 1$. What is left is the sum from $n = 0$ to $N - 1$ of $f[n]$, i.e. the sum of all pixel values. The zero frequency is the DC component, the constant level to which sines and cosines are added to form the image function.

b) In a real-valued image, $F[-k] = F^*[k]$, the complex conjugate of the value at the positive frequency.

c) $F[0]$ can only be complex if $f[n]$ has complex values. For real-valued images, $F[-0] = F^*[0]$, which can only be true if $F[0]$ is real. Furthermore, since $F[0]$ is the sum of pixel values, if $f[n]$ is real, the sum of its pixel values must be real too. But given a complex image $f[n]$, the sum of its values can be complex, and the complex conjugate symmetry does not hold.

6 Object description 5p

a) E.g. histograms and co-occurrence matrices. If they are normalized to sum to 1.

b) The following pairs are possible. 11, 12, 13, 21, 22, 23, 31, 32, 33. In row-column-order we arrange these into a matrix that is normalized (with 1/18):

$$\frac{1}{18}\begin{array}{|c|c|c|}\hline 0 & 3 & 4 \\ \hline 4 & 0 & 0 \\ \hline 4 & 3 & 0 \\ \hline \end{array}$$

Now uniformity is defined as $\sum_i \sum_j c_{ij}^2 = \frac{1}{18^2}(3^2 + 4^2 + 4^2 + 4^2 + 3^2) = 66/324$.

7 Local neighborhood operations 5p

a) This could be done in many ways ofcourse. The things we would like to see demonstrated are: Median filters preserve edges better than averaging filters. They also efficiently remove outliers (e.g. salt- and pepper noise). And median filters are less affected by border effects, but in the example below the corner pixels are in fact affected.

Image:

100	100	100	0	1	0	0	0	0
100	100	100	0	0	0	100	0	0
100	100	100	0	0	1	0	1	0
100	100	100	0	0	0	0	0	0

Averaging:

44	67	44	22	0	11	11	11	0
67	100	67	33	0	11	11	11	0
67	100	67	33	0	11	11	11	0
44	67	44	22	0	0	0	0	0

Median:

0	100	0	0	0	0	0	0	0
100	100	100	0	0	0	0	0	0
100	100	100	0	0	0	0	0	0
0	100	0	0	0	0	0	0	0

b) This question is easier than it seems. Enumerate pixels in a 3 x 3 neighborhood as z_1 to z_9. A simple non-linear filter would then be ($z_2 < z_5$ and $z_4 < z_5$ and $z_6 < z_5$ and $z_8 < z_5$) or ($z_2 > z_5$ and $z_4 > z_5$ and $z_6 > z_5$ and $z_8 > z_5$). So you loop over the image with a 3 x 3 window and perform this computation. If the logical expression is true, you output 1, else you output 0 for that pixel. Pad with 0s outside the image if necessary.

8 Segmentation 5p

a) It is easy to find a global threshold but the question is if this is a meaningful segmentation, since it would separate white pieces and white squares from black pieces and black squares. Finding s threshold for e.g. only the blac pieces would be a lot harder, considering the fact that there are only two visible peaks in the histogram and four classes of objects.

b) In the lecture notes a simple method is described where a threshold T is selected, e.g. 128 (average gray) Then the image is segmented using this threshold and after that the average background and foreground graylevels are computed. A new threshold is computed as the average of those two classes. The procedure is then iterated, with a new segmentation into foreground and background and so on. Until the threshold T does not change too much or not at all.