EE838 Assignment 5

Jonas Nikula
20176392

1 Algorithm explanation

1.1 AKAZE

AKAZE is an **Accelerated** version of KAZE. It's faster because it builds it's nonlinear scale space representation using a mathematic framework called Fast Explicit Diffusion. It has the following steps:
1. Building a Nonlinear Scale Space with Fast Explicit Diffusion
1.1 We first need the contrast factor λ, which we get from an algorithm defined in the AKAZE paper. That algorithm uses a gaussian smoothed input image.
1.2 The next step is to run the FED cycles, which are explicit diffusion schemes with varying time steps.
1.21 In AKAZE, the FED scheme is embedded in a fine to coarse pyramidal approach which increases computation speed and results in a set of filtered images.
2. Feature Detection
2.1 Compute the determinant of Hessian using the set of filtered images obtained from the previous step.
3. Feature Description
3.1 AKAZE uses a Modified-Local Difference Binary (M-LDB) descriptor.
3.11 The descriptor uses gradient and intensity information from the nonlinear scale space.

1.2 LIOP

1. Pre-processing, Feature Detection and Normalization
1.1 The input image is smoothed with a Gaussian filter.
1.2 An affine covariant detector is run to localize the features and estimate the affine shapes of their neighborhoods.
1.3 The detected regions are normalized to circular regions.
1.4 The image is smoothed a again with a Gaussian filter to remove noise from the normalization step.
2. Region Division
2.1 The output from the previous step is divided into subregions, according to their intensity.
3. Local Intensity Order Pattern Descriptor
3.1 The LIOP is calculated for each point in a subregion.
3.2 After that the LIOPs in each subregion are combined to create the descriptor.
3.21 Figure 1 has a graphical representation of the descriptor.

1.3 OIOP

OIOP is based on LIOP, and many of the steps are the same.
1. Intensity Order Based Patch Division
1.1. Like in LIOP, an affine covariant detector is used to localize a keypoint and estimate the affine shape of its neighborhood.
1.2. The resulting area is then normalized to a circle, and smoothed with a Gaussian filter. The resulting region is called a patch.
1.3. Like in LIOP, the patch is divided into subregions or bins according to their intensity.
1.4. A descriptor is created by constructing a low-level descriptor seperately for each bin, and concatenating the resulting descriptors.

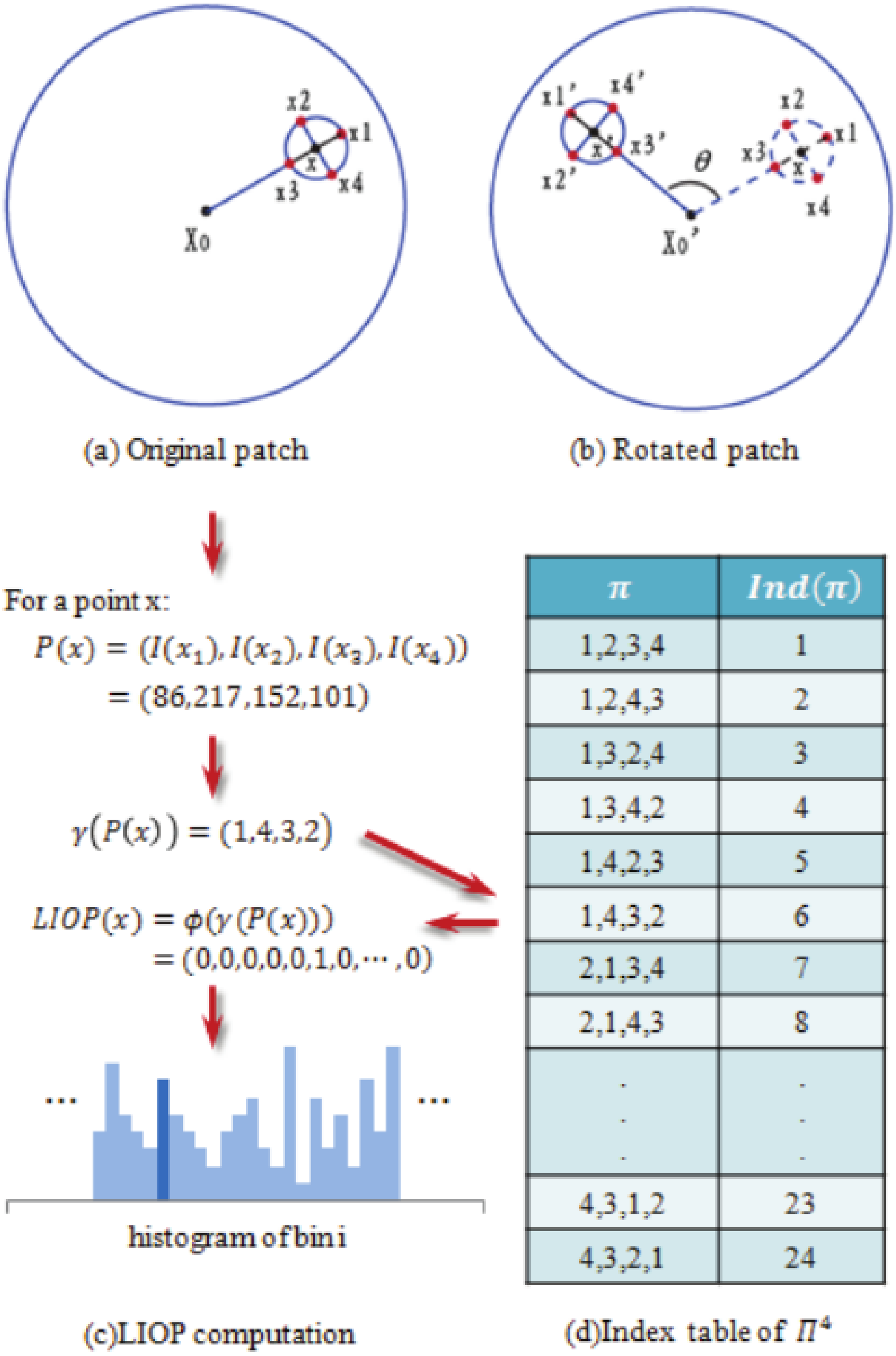

Figure 1: The LIOP descriptor, taken from the LIOP paper.

2. Local Intensity Order Pattern
2.1. Create the LIOP descriptor
3. Overall Intensity Order Pattern
3.1. The problem with LIOP is that because it uses neighbor points' intensity order, it is not robust to noise. OIOP intends to solve this by taking the overall intensity distribution in the patch to account.
3.2. This is done by quantizing the neighbor points' intensity, which effectively lessens the effect of noise on the intensity.
3.3. The quantizing can also be learning-based, which increases the adaptability of the algorithm.

2 Answers to given questions

2.1 Explain the difference between SIFT and A-KAZE for detecting local maxima.

SIFT tries to detect local maxima from linear Gaussian scale space. Linear scale-space has the disadvantage that the whole image is blurred. A-KAZE uses non-linear scale space, which preserves the strong edges of the image, while still removing noise.

2.2 Explain the difference between LIOP and OIOP descriptors.

The LIOP descriptor only takes into account the exact intensity between the sampling points. This makes it sensitive to noise, for example. The OIOP accounts for this by coarsely quantizing the intensity values of the points. In standard quantization, the quantization is based on a manually set constant. In learning based quantization the amount of quantization can be learned, thus increasing adaptability.

2.3 As we can see in (2), LIOP descriptor is rotation-invariant by itself. Explain the technical reason of the property.

The descriptor consists of the keypoint and its neighbors, ie. points around it. The neighbors are distributed equally in a circle around the keypoint. The first neighbor to be sample is along a line that goes from the center of the patch to the keypoint. As the neighbors are equally distributed, there are two neighbors on this line. The furthest from the center point is the first to be sampled. The other neighbors are sampled anticlockwise from the first point.

The descriptor is rotation invariant because this grouping of points retain their relation to each other and the patch no matter the rotation. This might be understood more clearly from the figure 1.