Problem Session #6

EE368/CS232
Digital Image Processing
1. Robustness of Harris Keypoints to Rotation and Scaling

Part A:
Apply a Harris corner detector and threshold the cornerness response so that about 450-850 Harris corners are detected. Please submit a result showing the detected Harris corners superimposed on the original image, and report the threshold that you choose. Describe which objects or regions in the image seem to generate large numbers of Harris corners. Hint: Use the MATLAB functions \texttt{cornermetric} and \texttt{imregionalmax}.

![Image of birds and detection results]
Part B:
Rotate the image in increments of 15 degrees, from 0 degrees to 360 degrees. For each rotated image, compute Harris corners using the same settings that you chose in part (a). Then, compute repeatability as follows.

• Set “number of feature matches” to be 0.
• For each Harris corner at \([x, y]\) in the original image:
  • Predict the position \([x_r, y_r]\) where \([x, y]\) should appear in the rotated image.
  • Search for a nearby Harris corner in the rotated image (coordinates \([x_o, y_o]\)) satisfying \(|x_o - x_r| \leq 2\) and \(|y_o - y_r| \leq 2\).
  • If such an \([x_o, y_o]\) is found, increment “number of feature matches” by 1.
• Compute repeatability as (number of feature matches) / (number of Harris corners in the original image).

Plot repeatability against rotation angle and comment on the Harris corner detector’s robustness against rotation.

\[
\text{imgRot} = \text{imrotate}(\text{img}, \text{angle}, \text{'bicubic'});
\]
Part B:

A detected keypoint \([x, y]\) in the original image.

Anti-clockwise rotation \(\theta\) to predicted keypoint \([x', y']\) in the rotated image.

\[
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} - \begin{bmatrix}
  x'_{\text{center}} \\
  y'_{\text{center}}
\end{bmatrix} = \begin{bmatrix}
  \cos(\theta) & \sin(\theta) \\
  -\sin(\theta) & \cos(\theta)
\end{bmatrix} \left( \begin{bmatrix}
  x \\
  y
\end{bmatrix} - \begin{bmatrix}
  x_{\text{center}} \\
  y_{\text{center}}
\end{bmatrix} \right)
\]

- **Predicted keypoint coordinates**
- **Center of rotated image**
- **Rotation matrix**
- **Detected keypoint in the original image**
- **Center of original image**
Part C:
Conduct an experiment analogous to part (b), but, instead of rotating the image, resize the image by the scaling factors $m^0, m^1, m^2, \ldots, m^8$, where $m = 1.2$. Compute Harris corners for each resized image. Please perform bicubic interpolation with MATLAB function `imresize`. By comparing the Harris corners of the original image and the corresponding Harris corners of the resized image, compute and plot repeatability against scaling factor (log scale may be most appropriate), and comment on the Harris corner detector’s robustness against scale changes.

```matlab
imgScale = imresize(img, scale, 'bicubic');
```
2. Robustness of SIFT Keypoints to Rotation and Scaling

Part A:
Apply a SIFT keypoint detector and adjust the peak and edge thresholds so that about 450-850 SIFT keypoints are detected. Please submit a result showing the detected SIFT keypoints superimposed on the original image, and report the thresholds you choose. Describe which objects or regions in the image seem to generate large numbers of SIFT keypoints.

% Set up MATLAB interface to VLFeat library
run('/path/to/library/vlfeat-0.9.17/toolbox/vl_setup.m');

% Extract SIFT keypoints on grayscale image
peakThresh = 4;
edgeThresh = 10;
[fc, dc] = vl_sift(single(gray_img), ...
    'PeakThresh', peakThresh, ...
    'EdgeThresh', edgeThresh);
We run the detector with peak threshold `peak_thresh` by

```matlab
f = vl_sift(I, 'PeakThresh', peak_thresh);
```

obtaining fewer features as `peak_thresh` is increased.

Detected frames for increasing peak threshold.
From top: `peak_thresh = {0, 10, 20, 30}`.

The `edge threshold` eliminates peaks of the DoG scale space whose curvature is too small (such peaks yield badly localized frames). For instance, consider the test image

```matlab
I = zeros(100,500);
for i=[10 20 30 40 50 60 70 80 90]
    I(50-round(i/3):50+round(i/3),i*5) = 1;
end
I = 2*pi*8^2 * vl_imsmooth(I,8);
I = single(255 * I);
```

A test image for the edge threshold parameter.

We run the detector with edge threshold `edge_thresh` by

```matlab
f = vl_sift(I, 'edgethresh', edge_thresh);
```
Eliminating low contrast responses

Peaks which are too short may have been generated by noise and are discarded. This is done by comparing the absolute value of the DoG scale space at the peak with the peak threshold \( t_p \) and discarding the peak if its value is below the threshold.

Eliminating edge responses

Peaks which are too flat are often generated by edges and do not yield stable features. These peaks are detected and removed as follows. Given a peak \( x, y, \sigma \), the algorithm evaluates the \( x,y \) Hessian of the DoG scale space at the scale \( \sigma \). Then the following score (similar to the Harris function) is computed:

\[
\frac{(\text{tr} D(x, y, \sigma))^2}{\text{det} D(x, y, \sigma)}, \quad D = \begin{bmatrix}
\frac{\partial^2 \text{DoG}}{\partial x^2} & \frac{\partial^2 \text{DoG}}{\partial x \partial y} \\
\frac{\partial^2 \text{DoG}}{\partial x \partial y} & \frac{\partial^2 \text{DoG}}{\partial y^2}
\end{bmatrix}.
\]

This score has a minimum (equal to 4) when both eigenvalues of the Jacobian are equal (curved peak) and increases as one of the eigenvalues grows and the other stays small. Peaks are retained if the score is below the quantity \((t_e + 1)(t_e + 1)/t_e\), where \( t_e \) is the edge threshold. Notice that this quantity has a minimum equal to 4 when \( t_e = 1 \) and grows thereafter. Therefore the range of the edge threshold is \([1, \infty)\).
Part B:
Using the procedure defined in Problem 2, plot repeatability versus rotation angle (in increments of 15 degrees, from 0 degrees to 360 degrees). Comment on the SIFT keypoint detector’s robustness against rotation and compare against the corresponding result for the Harris keypoint detector.
Robustness to Rotation: Cover-01.png

Robustness to Rotation: Cover-02.png

SIFT Keypoints

Harris Keypoints
Part C:
Using the same procedure defined in Problem 2, plot repeatability versus scaling factor (with the scaling factors $m^0, m^1, m^2, \ldots, m^8$, where $m = 1.2$). Comment on the SIFT keypoint detector’s robustness against scale changes and compare against the corresponding result for the Harris keypoint detector.
SIFT Keypoints

Harris Keypoints
3. Scale Selection for Determinant-of-Hessian

Consider a continuous-space version of the determinant-of-Hessian (DoH) keypoint detector. For an input image, the DoH response at scale $t$ is defined as follows:

\[
\text{det } H^t(x, y) = f^t_{xx}(x, y)f^t_{yy}(x, y) - \left(f^t_{xy}(x, y)\right)^2
\]

\[
f^t_{xx}(x, y) = \frac{\partial^2}{\partial x^2} g^t(x, y) * f(x, y)
\]

\[
f^t_{yy}(x, y) = \frac{\partial^2}{\partial y^2} g^t(x, y) * f(x, y)
\]

\[
f^t_{xy}(x, y) = \frac{\partial^2}{\partial y \partial x} g^t(x, y) * f(x, y)
\]

\[
g^t(x, y) = \frac{1}{2\pi t} \exp\left(-\frac{x^2 + y^2}{2t}\right)
\]
Part A:
Assume the input image is a Gaussian blob of scale $t_0 > 0$: $f(x, y) = 2\pi t_0 \cdot g^{t_0}(x, y)$. From the above definitions, derive and report an expression for the following objective functions:

$$H_A(t) = \det H'(0, 0)$$

$$H_B(t) = t \det H'(0, 0)$$

$$H_C(t) = t^2 \det H'(0, 0)$$

$$g^t(x, y) * f(x, y) = 2\pi t_0 \cdot g^t(x, y) * g^{t_0}(x, y) = 2\pi t_0 \cdot g^{t+t_0}(x, y)$$

$$f^{tx}_x(x, y) = \frac{\partial^2}{\partial x^2} \left( g^t(x, y) * f(x, y) \right) = 2\pi t_0 \frac{\partial^2}{\partial x^2} g^{t+t_0}(x, y)$$

$$f^{ty}_y(x, y) = 2\pi t_0 \frac{\partial^2}{\partial y^2} g^{t+t_0}(x, y)$$

$$f^{tx}_y(x, y) = f^{ty}_x(x, y) = 2\pi t_0 \frac{\partial^2}{\partial x \partial y} g^{t+t_0}(x, y)$$

$$\det H'(0, 0) = f^{xx}_x(0, 0) f^{yy}_y(0, 0) - \left( f^{xy}_x(0, 0) \right)^2$$
Part B:
Derive and report the scale values $t_A^*$, $t_B^*$, and $t_C^*$ that maximize $H_A(t)$, $H_B(t)$, and $H_C(t)$, respectively. Also report the maximal values. Please interpret the results.

\[
\begin{align*}
 t_A^* &= \ldots \\
 t_B^* &= \ldots \\
 t_C^* &= \ldots
\end{align*}
\]

Do $H_A$, $H_B$, and $H_C$ suitable for finding the scale $t_0$?

\[
\begin{align*}
 H_A\left(t_A^*\right) &= \ldots \\
 H_B\left(t_B^*\right) &= \ldots \\
 H_C\left(t_C^*\right) &= \ldots
\end{align*}
\]

Do maximum values of $H_A$, $H_B$, and $H_C$ depend on the scale $t_0$?