Machine Vision hw 1

Justin Hughes

January 2020

1 Problems 3-5

problem 3.

1.

Separating the filters into two 1D filters.
$\times$ indicates the outer product.

a.

$$\frac{1}{9}\begin{vmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{vmatrix} = \frac{1}{9}\begin{vmatrix}1\\ 1\\ 1\end{vmatrix} \times \begin{vmatrix}1 & 1 & 1\end{vmatrix}$$

b.

$$\begin{vmatrix}-1 & -2 & -1\\ 0 & 0 & 0\\ 1 & 2 & 1\end{vmatrix} = \begin{vmatrix}-1\\ 0\\ 1\end{vmatrix} \times \begin{vmatrix}1 & 2 & 1\end{vmatrix}$$

2.

First consider the separated filter with $F_{s1} \in \mathrm{R}^{\,k\times 1}$ and $F_{s2} \in \mathrm{R}^{\,1\times k}$. According to our assumption F_{s1} has computational complexity of $\mathrm{O}(k \times 1 \times H \times W)$ and F_{s2} has computational complexity of $\mathrm{O}(1 \times k \times H \times W)$ with $\mathrm{I} \in \mathrm{R}^{H\times W}$. What we can conclude from this is the computational complexity for the separated filter is $2 \times O(kHW)$ since we are using convolution of two separate filters over the image.

problem 4.

Difference between LoG and Laplacian filters. What makes LoG better.
The Lacplacian filter may detect edges but it also may detect flase edges. This is due to the Laplacian filter being more susceptible to noise. Because it is so sensitive to noise, we reduce the image down with the Gaussian filter.

problem 5.

Derivation:

$$E(u,v,w) = \sum [I(x+u, y+v, t+w) - I(x,y,t)]^2$$

$$\sum [I(x,y,z) + I_x u + I_y v + I_t w - I(x,y,t)]^2$$

Since we are making small movements we discarded large order values and only consider first order. We then have:

$$\sum [I_x u + I_y v + I_t w]^2$$

Consider the Taylor series.

$$\sum (I_x u)^2 + (I_y v)^2 + (I_t w)^2 + 2(I_x u I_y v) + 2(I_y v I_t w) + 2(I_t w I_x u)$$

The 3D structor tensor:

$$\begin{vmatrix} u & v & w \end{vmatrix} \begin{vmatrix} I_x^2 & I_x I_y & I_x I_z \\ I_y I_x & I_y^2 & I_y I_z \\ I_z I_x & I_z I_y & I_z^2 \end{vmatrix} \begin{vmatrix} u \\ v \\ w \end{vmatrix}$$

This shape is a blob. The criteria we would need in order to find a corner would be determining if the intensity changes as the direction changes. So if we move in any direction and get drastic changes in intensity we should be able to detect a corner.

Different structures:

When we increase any of the given eigenvalues, we will result in a shape that is elongated in that direction. Having high eigenvalues will result in more interesting points in the image. When moving along the object in a video scenario, we get a corner when intensity drastically changes in all directions. If we were to consider one of the eigenvalues as constant while the others increase, we would get a 2D corner. The way I interpret a 3D corner is a 2D corner with excess 3D features in all directions.