2D Vision and Deep Learning
Assignment 8

Amadeus Seibert
Marcel Hehn
Salem Zin Iden Naser

25. Januar 2021

1 Exercise 1

z.Z.: $\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2 = \left(\frac{\partial f}{\partial r}\right)^2 + \frac{1}{r^2}\left(\frac{\partial f}{\partial \Theta}\right)^2$

$\frac{\partial r}{\partial x} = \frac{2x}{2\sqrt{x^2+y^2}} = \cos(\Theta)$

$\frac{\partial r}{\partial y} = \frac{2y}{2\sqrt{x^2+y^2}} = \sin(\Theta)$

$\frac{\partial \Theta}{\partial x} = \frac{1}{1+(\frac{y}{x})^2} + \frac{-y}{x^2} = \frac{-y}{x^2+y^2} = \frac{-\sin(\Theta)}{r}$

$\frac{\partial \Theta}{\partial y} = \frac{1}{1+(\frac{y}{x})^2} + \frac{1}{x} = \frac{x}{x^2+y^2} = \frac{\cos(\Theta)}{r}$

$$\begin{aligned}
\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2 &= \left(\frac{\partial f}{\partial r}\cdot\frac{\partial r}{\partial x} + \frac{\partial f}{\partial \Theta}\cdot\frac{\partial \Theta}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial r}\cdot\frac{\partial r}{\partial y} + \frac{\partial f}{\partial \Theta}\cdot\frac{\partial \Theta}{\partial y}\right)^2 \\
&= \left(\frac{\partial f}{\partial r}\right)^2 \cdot (\left(\frac{\partial r}{\partial x}\right)^2 + \left(\frac{\partial r}{\partial y}\right)^2) + \left(\frac{\partial f}{\partial \Theta}\right)^2 \cdot (\left(\frac{\partial \Theta}{\partial x}\right)^2 + \left(\frac{\partial \Theta}{\partial y}\right)^2) \\
&+ 2\left(\frac{\partial f}{\partial r}\cdot\frac{\partial f}{\partial \Theta}\right)\cdot(\frac{\partial r}{\partial x}\cdot\frac{\partial \Theta}{\partial x} + \frac{\partial r}{\partial y}\cdot\frac{\partial \Theta}{\partial y}) \\
&= \left(\frac{\partial f}{\partial r}\right)^2 \cdot (\cos^2(\Theta) + \sin^2(\Theta)) + \left(\frac{\partial f}{\partial \Theta}\right)^2 \cdot (\frac{\sin^2(\Theta)}{r^2} + \frac{\cos^2(\Theta)}{r^2}) \\
&+ 2\left(\frac{\partial f}{\partial r}\cdot\frac{\partial f}{\partial \Theta}\right)\cdot(\frac{-\cos(\Theta)\sin(\Theta)}{r} + \frac{\sin(\Theta)\cos(\Theta)}{r}) \\
&= \left(\frac{\partial f}{\partial r}\right)^2 \cdot 1 + \left(\frac{\partial f}{\partial \Theta}\right)^2 \cdot \frac{1}{r^2} + 2\left(\frac{\partial f}{\partial r}\cdot\frac{\partial f}{\partial \Theta}\right)\cdot 0 \\
\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2 &= \left(\frac{\partial f}{\partial r}\right)^2 + \frac{1}{r^2}\left(\frac{\partial f}{\partial \Theta}\right)^2 \square
\end{aligned}$$

2 Exercise 2

The filter size is 2x2 and the stride is 2, which means that each input pixel was filtered together with 3 other pixels ONCE. The pixels that aren't the max of their tuple of 4 pixels have a gradient of 0 because they don't pass the forward filter and therefore don't take part in the error, so we can use 0 as the initial value. For each tuple of 4 input pixels there will be a maxima, which a subset of the pixels will be equal to, all those pixels will have their gradient set to the gradient of the pixel one layer deeper, which value was determined by this subset. Using this method we don't change more then needed avoiding unintended behaviour.

3 Exercise 3

3.1

Assuming we have a padding

$$
\begin{array}{|c|c|c|c|c|c|}
\hline
 & & & & & \\ \hline
 & & & & & \\ \hline
 & & x_1 & & & \\ \hline
 & & & & & \\ \hline
 & & & & & \\ \hline
 & & & & & \\ \hline
\end{array}
*
\begin{array}{|c|c|c|}
\hline
w_1 & w_2 & w_3 \\ \hline
w_4 & w_5 & w_6 \\ \hline
w_7 & w_8 & w_9 \\ \hline
\end{array}
\rightarrow
\begin{array}{|c|c|c|c|c|c|}
\hline
 & & & & & \\ \hline
 & y_1 & y_2 & y_3 & & \\ \hline
 & y_4 & x_5 & y_6 & & \\ \hline
 & y_7 & y_8 & y_9 & & \\ \hline
 & & & & & \\ \hline
 & & & & & \\ \hline
\end{array}
$$

3.2

$y_1 = w_9 \cdot x_1 + C_1$
$y_2 = w_8 \cdot x_1 + C_2$
$y_3 = w_7 \cdot x_1 + C_3$
$y_4 = w_6 \cdot x_1 + C_4$
$y_5 = w_5 \cdot x_1 + C_5$
$y_6 = w_4 \cdot x_1 + C_6$
$y_7 = w_3 \cdot x_1 + C_7$
$y_8 = w_2 \cdot x_1 + C_8$
$y_9 = w_1 \cdot x_1 + C_9$

3.3

Because C_i isn't a function of x_1 it can be treated as constant to x_1.
$\frac{\partial v}{\partial x_1} = [w_9, w_8, w_7, w_6, w_5, w_4, w_3, w_2, w_1]^T$

3.4

x_1 only affects y_1 to y_9, because of this the error also only back-propagates through those. While the weights determine how much of the error was caused by x_1, in the previous subsections we showed that x_i causes an error for y_{9-i} through weight w_i, thus we get exactly the convolution filter w'.

3.5

Back-propagating through this convolution layer seems like an excellent choice, because it would be fast and easy, while casting no additional space as we already have the filter w.