Homework 1: Computer Vision and Image Analysis - Solutions
4 Parts, 3 Pages

Interdisciplinary Space Master (ISM)/Master in Information and Computer Science (MICS)
Interdisciplinary Centre for Security, Reliability and Trust
Computer Vision, Imaging, and Machine Intelligence (CVI^2) group

Course Responsible: Prof. Djamila Aouada
TA: Dr. Konstantinos Papadopoulos
djamila.aouada@uni.lu; konstantinos.papadopoulos@uni.lu

October 20, 2020

Part I: Image filtering

Considering an image denoising problem, let $\mathtt{x}$ be a noise-free $m \times m$ image degraded by an additive white Gaussian noise $\mathtt{w}$. The observed corrupted image $\mathtt{y}$ is given by:

$$\mathtt{y} = \mathtt{x} + \mathtt{w}. \tag{1}$$

The Gaussian filter recovers the original image $\mathtt{x}$ by a linear filtering that replaces the noisy intensity value $\mathtt{y}_\mathtt{p}$ at each pixel location $\mathtt{p}$ with a weighted average of the neighboring pixels $\mathtt{q} \in \mathcal{N}(\mathtt{p})$, such that:

$$\hat{\mathtt{x}}_\mathtt{p} = \frac{\sum_{\mathtt{q}\in\mathcal{N}(\mathtt{p})} f(\mathtt{p},\mathtt{q}) \cdot \mathtt{y}_\mathtt{q}}{\sum_{\mathtt{q}\in\mathcal{N}(\mathtt{p})} f(\mathtt{p},\mathtt{q})}. \tag{2}$$

1. Write the analytical expression of the Gaussian kernel f with standard deviation σ_f?
   Answer: **(see slides 8 and 9 from the first lecture)**
   $\mathtt{x} \in \mathbb{R}^{m\times m}$, $\mathtt{p},\mathtt{q} \in \mathbb{R}^2$, $\mathtt{y} = \mathtt{x} + \mathtt{w}$ where $\mathtt{w}$ is additive noise.
   From $\mathtt{y}$, we want to recover $\mathtt{x}$ or estimate $\hat{\mathtt{x}}$.
   $\mathtt{p}$: location in the "domain"; let $\mathtt{p} = \begin{pmatrix} i \\ j \end{pmatrix} \in \mathbb{R}^2$

$\mathtt{x}_\mathtt{p} \equiv \mathtt{x}(\mathtt{p}) \equiv \mathtt{x}(i,j)$: value at $\mathtt{p}$ in the range of the image. We have:

$$\hat{\mathtt{x}}_\mathtt{p} = \frac{\sum\limits_{\mathtt{q}\in\mathcal{N}(\mathtt{p})} f(\mathtt{p},\mathtt{q})\cdot \mathtt{y}_\mathtt{q}}{\sum\limits_{\mathtt{q}\in\mathcal{N}(\mathtt{p})} f(\mathtt{p},\mathtt{q})} = \frac{1}{K}\sum_{\mathtt{q}\in\mathcal{N}(\mathtt{p})} f(\mathtt{p},\mathtt{q})\cdot \mathtt{y}_\mathtt{q},$$

where K is a constant normalization factor. However, f is not normalized; we can define the normalized Gaussian kernel $f_n = \frac{f}{K}$

$$f_n(\mathtt{p},\mathtt{q}) = \frac{1}{2\pi\sigma_f^2} exp\left(-\frac{||\mathtt{p}-\mathtt{q}||}{2\sigma_f^2}\right) \tag{1}$$

Using notation from the first lecture (**Slide 58**):

$$\mathtt{p} = \begin{pmatrix} i \\ j \end{pmatrix}; \quad \mathtt{q} = \begin{pmatrix} k \\ l \end{pmatrix}$$

$$\text{Thus: } ||\mathtt{p}-\mathtt{q}||^2 = \left|\left| \begin{pmatrix} i-k \\ j-l \end{pmatrix} \right|\right|^2 = (i-k)^2 + (j-l)^2 \tag{2}$$

$$f_n(\mathtt{p},\mathtt{q}) \equiv f_n(\mathtt{p}-\mathtt{q}) = f_n(i-k, j-l) \tag{3}$$

From (1), (2) and (3) and for $i-k=i'$ and $j-l=i'$ we get:

$$f_n(i',j') = \frac{1}{2\pi\sigma_f^2} exp\left(-\frac{i'^2+j'^2}{2\sigma_f^2}\right)$$

2. Show that (2) may be rewritten using convolutions.
Answer:

$$\hat{\mathtt{x}}_\mathtt{p} = \hat{\mathtt{x}}(i,j)\hat{\mathtt{x}}_\mathtt{p} = \sum_{\mathtt{q}\in\mathcal{N}(\mathtt{p})} f_n(\mathtt{p},\mathtt{q})\cdot \mathtt{y}_\mathtt{q}$$

Using notation from the first lecture (**Slide 47**), we may write

$$\begin{aligned}
\hat{\mathtt{x}}(i,j) &= \sum_k \sum_l f_n(i-k, j-l)\mathtt{y}(k,l) \\
&= \sum_{i'} \sum_{j'} f_n(i',j')\mathtt{y}(i-i', j-j') \\
&= (f_n * \mathtt{y})(i,j) \\
&= \frac{1}{K}(f * \mathtt{y})(i,j)
\end{aligned}$$

3. Assuming the pixels $\mathtt{y_p}$ are i.i.d. with a standard deviation $\sigma_{\mathtt{y}}$, compute the noise reduction $r = \frac{\sigma_{\hat{\mathtt{x}}}}{\sigma_{\mathtt{y}}}$ where $\sigma^2_{\hat{\mathtt{x}}}$ is the variance of the filtered image $\hat{\mathtt{x}}$.
Answer: We have $var(\hat{\mathtt{x}}(i,j)) = \sigma^2_{\hat{\mathtt{x}}}$. Therefore:

$$\begin{aligned}
\hat{\mathtt{x}}(i,j) &= \sum\sum f_n(i',j')\mathtt{y}(i-i',j-j') \\
&= \sum\sum f_n^2(i',j') \underbrace{var[\mathtt{y}(i-i',j-j')]}_{\sigma^2_{\mathtt{y}}} \\
&= \sigma^2_{\mathtt{y}} \sum_{i'}\sum_{j'} f_n^2(i',j') \\
&\cong \sigma^2_{\mathtt{y}} \int\int \frac{1}{(2\pi\sigma_f^2)} \left[exp\left(-\frac{i'^2}{2\sigma_f^2}\right)\left[exp\left(-\frac{j'^2}{2\sigma_f^2}\right)\right]^2 di'dj' \\
&= \frac{\sigma^2_{\mathtt{y}}}{4\pi^2\sigma_f^4} \int exp\left(-\frac{i'^2}{2\sigma_f^2}\right)di' \int exp\left(-\frac{j'^2}{2\sigma_f^2}\right)dj'
\end{aligned}$$

$$\text{Given: } \int e^{-a(x+b)^2}dx = \sqrt{\frac{\pi}{a}} \Rightarrow exp\left(-\frac{i'^2}{2\sigma_f^2}\right)di' = \sqrt{\pi\sigma_f^2}$$

$$\text{Thus we find: } \sigma^2_{\hat{\mathtt{x}}} = \left(\frac{\pi\sigma_f^2}{4\pi^2\sigma_f^4}\right)\sigma^2_{\mathtt{y}} \Rightarrow r = \frac{\sigma_{\hat{\mathtt{x}}}}{\sigma_{\mathtt{y}}} = \frac{1}{2\sqrt{\pi}\sigma_f}$$

4. Give a suitable integer-valued convolution mask $\mathtt{f}$ of size (5×5) that approximates Gaussian filter f with a standard deviation $\sigma_f = 1.4$.
Answer: $f = \frac{1}{159}\begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$

Part II: Separable kernels

Show whether the following convolution kernel are separable or not. If separable, indicate the separations.

$$\mathtt{f} = \begin{bmatrix} 2 & 0 & 6 \\ 0 & 1 & 0 \\ 1 & 0 & 3 \end{bmatrix}$$

Answer: A matrix is separable if each row is a scalar multiple of the same row (the rank is 1). Thus, it is not separable.

Part III: Image features

Considering the image $\mathtt{y}$, an interest point detector typically computes the matrix $\mathtt{M}_\mathtt{p}$ which captures the changes of pixel intensities in a neighborhood $\mathcal{N}(\mathtt{p})$ of each pixel location $\mathtt{p}$. A pixel intensity change at location $\mathtt{p}$ can be computed by filtering $\mathtt{y}$ with a filter $\mathtt{g}$.

1. Give the expressions of a (3×3) filter $\mathtt{g}$ and the expression of $\mathtt{M}_\mathtt{p}$, each time explaining your notation, for:

   (a) the Harris detector
   Answer: Let

   $$\mathtt{y}_i = \frac{\partial \mathtt{y}}{\partial i}; \mathtt{y}_j = \frac{\partial \mathtt{y}}{\partial i};$$

   $$\mathtt{M}_\mathtt{p} = \begin{bmatrix} \mathtt{y}_i^2 & \mathtt{y}_i\mathtt{y}_j \\ \mathtt{y}_i\mathtt{y}_j & \mathtt{y}_j^2 \end{bmatrix}$$

   Gradient (smoothed) e.g. Sobel:

   $$\mathtt{y}_i = \mathtt{g} * \mathtt{y}$$

   with

   $$\mathtt{g} = \underbrace{\frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}}_{\text{Gaussian}} * \underbrace{\frac{1}{2}\begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}}_{\text{Gradient}} = \frac{1}{8}\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

   $$\mathtt{y}_j = \mathtt{g} * \mathtt{y}$$

   with

   $$\mathtt{g} = \underbrace{\frac{1}{4}\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}}_{\text{Gaussian}} * \underbrace{\frac{1}{2}\begin{bmatrix} 1 & 0 & -1 \end{bmatrix}}_{\text{Gradient}} = \frac{1}{8}\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

   You could use a different smoothing kernel. These are only examples.

   (b) the Hessian detector
   Answer: Let

   $$\mathtt{y}_{ii} = \frac{\partial^2 \mathtt{y}}{\partial i^2}; \mathtt{y}_{jj} = \frac{\partial^2 \mathtt{y}}{\partial j^2}$$

   $$\mathtt{M}_\mathtt{p} = \begin{bmatrix} \mathtt{y}_{ii} & \mathtt{y}_i\mathtt{y}_j \\ \mathtt{y}_i\mathtt{y}_j & \mathtt{y}_{jj} \end{bmatrix}$$

$$\mathtt{y}_i\mathtt{y}_j = \frac{\partial^2 \mathtt{y}}{\partial i \partial j} = \mathtt{g} * \mathtt{y}$$

with

$$\mathtt{g} = \frac{1}{2}\begin{bmatrix}1\\0\\-1\end{bmatrix} * \frac{1}{2}\begin{bmatrix}1 & 0 & -1\end{bmatrix} = \frac{1}{4}\begin{bmatrix}1 & 0 & -1\\0 & 0 & 0\\-1 & 0 & 1\end{bmatrix}$$

$$\mathtt{y}_{ii} = \mathtt{g} * \mathtt{y}$$

with

$$\mathtt{g} = \begin{bmatrix}1\\-1\end{bmatrix} * \begin{bmatrix}1\\-1\end{bmatrix} * \frac{1}{4}\begin{bmatrix}1 & 2 & 1\end{bmatrix} = \underbrace{\begin{bmatrix}1\\-2\\1\end{bmatrix}}_{\substack{\text{Without}\\\text{smoothing}}} * \underbrace{\frac{1}{4}\begin{bmatrix}1 & 2 & 1\end{bmatrix}}_{\text{Gaussian}} = \frac{1}{4}\begin{bmatrix}1 & 2 & 1\\-2 & -4 & -2\\1 & 2 & 1\end{bmatrix}$$

$$\mathtt{y}_{jj} = \mathtt{g} * \mathtt{y}$$

with

$$\mathtt{g} = \frac{1}{4}\begin{bmatrix}1\\2\\1\end{bmatrix} * \begin{bmatrix}1 & -2 & 1\end{bmatrix} = \frac{1}{4}\begin{bmatrix}1 & -2 & 1\\2 & -4 & 2\\1 & -2 & 1\end{bmatrix}$$

2. Let $\lambda_{\mathtt{p}}^1$ and $\lambda_{\mathtt{p}}^2$ denote the eigenvalues of $\mathtt{M}_{\mathtt{p}}$. What kind of image feature is present at $\mathtt{p}$ when:

   (a) $0 < \lambda_{\mathtt{p}}^1 \ll \lambda_{\mathtt{p}}^2$ of the Harris detector?
   Answer: Edge.

   (b) $\lambda_{\mathtt{p}}^1 \approx \lambda_{\mathtt{p}}^2 \gg 0$ of the Hessian detector?
   Answer: Corner.

   (c) $\lambda_{\mathtt{p}}^1 = \lambda_{\mathtt{p}}^2 = 0$ of the Hessian detector?
   Answer: Noise.

   (d) $\lambda_{\mathtt{p}}^1 = \lambda_{\mathtt{p}}^2 = 0$ of the Harris detector?
   Answer: Uniform surface.

   (e) $0 < \lambda_{\mathtt{p}}^1$ or $0 < \lambda_{\mathtt{p}}^2$ of the Harris detector?
   Answer: Depending on the values, it can be either edge, corner or uniform surface.

3. What are the invariance properties of: (a) the Harris detector? (b) the Hessian detector?
   Answer: In their basic form, they are both robust to image plane rotations, illumination changes and some extent noise. They are not scale invariant.

4. The SIFT descriptor is known for its scale invariance property where the idea is to find local extrema in the scale space parametrized by the standard deviation σ_f, such that:

$$\mathtt{z} = (\mathtt{f}_{\sigma_f} - \mathtt{f}_{k\sigma_f}) * \mathtt{y} \tag{3}$$

where k is a fixed scalar factor, and $\mathtt{f}_{\sigma_f}$ is a Gaussian kernel with variance σ_f.

(a) Why is the kernel $\mathtt{f}$ chosen to be Gaussian?
Answer: The Difference of Gaussians function will have strong responses along edges by eliminating low frequency regions.

(b) Show that $\mathtt{z}$ is an approximation of the Laplacian of $\mathtt{y}$.
Answer: The Difference of Gaussians (DoG) is given by:

$$\mathtt{z} = (\mathtt{f}_{\sigma_f} - \mathtt{f}_{k\sigma_f}) * \mathtt{y} = L_{\sigma_f} - L_{k\sigma_f} \quad \text{where } L_{\sigma_f} = \mathtt{f}_{\sigma_f} * \mathtt{y}$$

The scale-normalized Laplacian of Gaussians (LoG) image representation is given by:

$$v = \sigma^2 \nabla^2 L_\sigma$$

The DoG and LoG can be related through the use of the (heat) diffusion equation,

$$\frac{\partial L_\sigma}{\partial \sigma} = \sigma \nabla^2 L_\sigma$$

The above diffusion equation can be approximated as follows,

$$\sigma \nabla^2 L_\sigma = \frac{\partial L_\sigma}{\partial \sigma} = \lim_{k \to 0} \frac{L_\sigma - L_{k\sigma}}{\sigma - k\sigma} \approx \frac{L_\sigma - L_{k\sigma}}{\sigma - k\sigma}$$

Finally, rearranging the above equation yields,

$$\begin{aligned}(\sigma - k\sigma)\sigma \nabla^2 L_\sigma &\approx L_\sigma - L_{k\sigma} \\ (1-k)\sigma^2 \nabla^2 L_\sigma &\approx L_\sigma - L_{k\sigma} \\ (1-k)\sigma^2 v &\approx z\end{aligned}$$

As can be seen, the DoG approximates the scale normalized LoG up to a (negligible) multiplicative constant, $1 - k$, that is present at all scales.

(c) How is the factor k typically chosen?
Answer: The factor k is chosen such that it divides each octave of scale space into an integer number, s, of intervals, so $k = 2^{\frac{1}{s}}$.

Part IV: Image matching

Given two sets of descriptors $\{\mathtt{v}_1^i\}$ and $\{\mathtt{v}_2^j\}$ extracted from two images to be matched, a distance d is computed between $\mathtt{v}_1^i$ and $\mathtt{v}_2^j$.

1. Give the expression of $d(\mathtt{v}_1^i, \mathtt{v}_2^j)$ if

   (a) d is the sum of squared differences (SSD)
   Answer: $d(\mathtt{v}_1^i, \mathtt{v}_2^j) = \sum_{i,j}(\mathtt{v}_1^i - \mathtt{v}_2^j)^2$

   (b) d is the cross correlation (CC)
   Answer: $d(\mathtt{v}_1^i, \mathtt{v}_2^j) = \sum_{i,j}(\mathtt{v}_1^i \mathtt{v}_2^j)$

2. What is the relationship between the two distances?
   Answer: SSD:

$$\begin{aligned} d(\mathtt{v}_1^i, \mathtt{v}_2^j) &= \sum_{i,j}(\mathtt{v}_1^i - \mathtt{v}_2^j)^2 \\ &= \sum_{i,j}(\mathtt{v}_1^i)^2 + \sum_{i,j}(\mathtt{v}_2^j)^2 - 2\sum_{i,j}(\mathtt{v}_1^i \mathtt{v}_2^j) \\ &= \sum_{i,j}(\mathtt{v}_1^i)^2 + \sum_{i,j}(\mathtt{v}_2^j)^2 - 2\mathrm{CC} \end{aligned}$$

   From the above, it can easily be shown that by minimizing the SSD, the CC is maximized.

3. RANSAC is used for a robust matching that discards outliers. The algorithm starts by matching s randomly selected elements of $\{\mathtt{v}_1^i\}$ are to the set $\{\mathtt{v}_2^j\}$. The operation is repeated T times keeping the largest number of inliers. If e is the ratio of outliers, what is T so that, with probability p, at least one random sample set is free from outliers?
   Answer: **(Slide 60 from the second lecture)** We choose T so that, with probability p, at least one random sample set from the total number of sampled points s is free from outliers:

$$\begin{aligned} 1 - p = 1 - ((1-e)^s)^T \Rightarrow \\ log(1-p) = T log(1-(1-e)^s) \Rightarrow \\ T = \frac{log(1-p)}{log(1-(1-e)^s)} \end{aligned}$$

Solution for Homework 2: Computer Vision and Image Analysis
2 Parts, 2 Pages

Interdisciplinary Space Master (ISM)/Master in Information and Computer Science (MICS)
Interdisciplinary Centre for Security, Reliability and Trust
Computer Vision, Imaging, and Machine Intelligence (CVI^2) group

Course Responsible: Prof. Djamila Aouada
TA: Dr. Anis Kacem
djamila.aouada@uni.lu; anis.kacem@uni.lu

06 October 2020
Due date: 21 October 2020 at 13:00

Part I: Maximum Likelihood Estimation

Suppose we are given a set of images of fruits with four categories: *banana, orange, apple, and strawberry*. Each image contains only one fruit from these categories. We assume that the probability for an image to be categorized into one of these fruits is a function parametrized by a scalar parameter $0 \leq \theta \leq 1$:

- The probability to be categorized to *banana* is $\frac{2\theta}{3}$
- The probability to be categorized to *orange* is $\frac{\theta}{3}$
- The probability to be categorized to *apple* is $\frac{2(1-\theta)}{3}$
- The probability to be categorized to *strawberry* is $\frac{(1-\theta)}{3}$

We assume that the categories of the images are modeled using a discrete random variable X.

1. Write the discrete random variable and its probability mass function.

   The discrete random variable and its probability mass function are given by:

X	*0 (banana)*	*1 (orange)*	*2 (apple)*	*3 (strawberry)*
P(X)	$\frac{2\theta}{3}$	$\frac{\theta}{3}$	$\frac{2(1-\theta)}{3}$	$\frac{(1-\theta)}{3}$

2. Suppose that the set of images contains 10 images and we know the category of each of them. The categories of these 10 images are as follows, (*strawberry, banana, apple, orange, strawberry, apple, orange, banana, apple, orange*). Write the likelihood function.

   The events of the discrete random variable X are $(3, 0, 2, 1, 3, 2, 1, 0, 2, 1)$. The Likelihood function is given by:

$$\begin{aligned} L(\theta) =& P(X=3)P(X=0)P(X=2)P(X=1)P(X=3) \\ &\times P(X=2)P(X=1)P(X=0)P(X=2)P(X=1) \end{aligned} \tag{1}$$

   Substituting from the probability distribution given above, we have:

$$L(\theta) = \prod_{i=1}^{n} P\left(X_i \mid \theta\right) = \left(\frac{2\theta}{3}\right)^2 \left(\frac{\theta}{3}\right)^3 \left(\frac{2(1-\theta)}{3}\right)^3 \left(\frac{1-\theta}{3}\right)^2 \tag{2}$$

3. What is the value of the maximum likelihood estimate of θ?

   Clearly, the likelihood function $L(\theta)$ is not easy to maximize. Let us look at the log-likelihood function:

$$\begin{aligned} l(\theta) = \log L(\theta) &= \sum_{i=1}^{n} \log P\left(X_i \mid \theta\right) \\ &= 2\left(\log\frac{2}{3} + \log\theta\right) + 3\left(\log\frac{1}{3} + \log\theta\right) + 3\left(\log\frac{2}{3} + \log(1-\theta)\right) \\ &+ 2\left(\log\frac{1}{3} + \log(1-\theta)\right) \\ &= C + 5\log\theta + 5\log(1-\theta) \end{aligned} \tag{3}$$

   where C is a constant which does not depend on θ. It can be seen that the log-likelihood function is easier to maximize compared to the likelihood function. Let the derivative of $l(\theta)$ with respect to θ be zero:

$$\frac{dl(\theta)}{d\theta} = \frac{5}{\theta} - \frac{5}{1-\theta} = 0 \tag{4}$$

   and the solution gives us the MLE, which is $\theta = 0.5$

Part II: Univariate Linear Regression

We recall that the regression line for a set of n data points is given by $y_i = ax_i + b$. In a univariate linear regression case, a and b are two scalar values in $\mathbb{R}$.

1. Using the least square method, demonstrate that for a univariate case,

$$a = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2} \tag{5}$$

and

$$b = \frac{1}{n}\left(\sum_{i=1}^{n} y_i - a\sum_{i=1}^{n} x_i\right) \tag{6}$$

In a linear regression setting, we would like to minimize the sum of squares error E,

$$E = \sum_{i=1}^{n} (y_i - \hat{y})^2 = \sum_{i=1}^{n} (y_i - b - ax_i)^2 \tag{7}$$

Then, E will be minimized at the values of b and a for which $\frac{\partial E}{\partial b} = 0$ and $\frac{\partial E}{\partial a} = 0$. The first of these conditions is,

$$\frac{\partial E}{\partial b} = \sum_{i=1}^{n} -2\,(y_i - b - ax_i) = 2\left(nb + a\sum_{i=1}^{n} x_i - \sum_{i=1}^{n} y_i\right) = 0 \tag{8}$$

which, if we divide through by 2 and solve for b, becomes simply,

$$b = \frac{1}{n}\left(\sum_{i=1}^{n} y_i - a\sum_{i=1}^{n} x_i\right) \tag{9}$$

For notation simplicity, let us consider $\bar{y} = \frac{1}{n}(\sum_{i=1}^{n} y_i)$ and $\bar{x} = \frac{1}{n}(\sum_{i=1}^{n} x_i)$ then,

$$b = \bar{y} - a\bar{x} \tag{10}$$

The second condition for minimizing E is,

$$\frac{\partial E}{\partial a} = \sum_{i=1}^{n} -2x_i\,(y_i - b - ax_i) = \sum_{i=1}^{n} -2\left(x_i y_i - bx_i - ax_i^2\right) = 0 \tag{11}$$

If we substitute the expression for b from (10) into (11), then we get,

$$\sum_{i=1}^{n} \left(x_i y_i - x_i \bar{y} + a x_i \bar{x} - a x_i^2 \right) = 0 \tag{12}$$

We can separate this into two sums,

$$\sum_{i=1}^{n} \left(x_i y_i - x_i \bar{y} \right) - a \sum_{i=1}^{n} \left(x_i^2 - x_i \bar{x} \right) = 0 \tag{13}$$

which gives directly,

$$a = \frac{\sum_{i=1}^{n} \left(x_i y_i - x_i \bar{y} \right)}{\sum_{i=1}^{n} \left(x_i^2 - x_i \bar{x} \right)} \tag{14}$$

by replacing $\bar{x}$ and $\bar{y}$ by their corresponding values, we can obtain

$$a = \frac{\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \frac{1}{n} \sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{n} x_i \frac{1}{n} \sum_{i=1}^{n} x_i} = \frac{n \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i \right)^2} \tag{15}$$

2. Consider the linear regression problem of the following data points: $\{(x_i, y_i)\}_{i=1}^{4} = \{(-1, 0), (0, 2), (1, 4), (2, 5)\}$.
Find the least square regression line $y_i = a x_i + b$.

We have 4 data points. Consider the following table:

data point	x_i	y_i	$x_i y_i$	x_i^2
1	-1	0	0	1
2	0	2	0	0
3	1	4	4	1
4	2	5	10	4
total	$\sum x_i = 2$	$\sum y_i = 11$	$\sum x_i y_i = 14$	$\sum x_i^2 = 6$

We now use the above formula to calculate a and b as follows,

$a = (4 * 14 - 2 * 11)/(4 * 6 - 2 * 2) = 17/10 = 1.7$

$b = (1/4)(11 - 1.7 * 2) = 1.9$

The regression line is finally given by,

$$y_i = 1.7 x_i + 1.9 \tag{16}$$

3. What is the estimated value of y if $x = 5$?

$y = 1.7 * 5 + 1.9 = 10.4$

4. Suppose that we have a short video of a car with its recorded speed for every image of the video. Further assume that the video contains only 7 frames (images) that are captured every 1 second. The speed of the car in that video was as follows,

Time (h:m:s)	7:29:15	7:29:16	7:29:17	7:29:18	7:29:19	7:29:20	7:29:21
Speed (km/h)	117	119	120	123	126	130	132

Provide an estimation for the speed of the car at 7:30:00 using a linear regression model.

The data points in Table 4 can be rewritten in the following from

t	0	1	2	3	4	5	6
Speed (km/h)	117	119	120	123	126	130	132

where t is an integer representing seconds starting from 7:29:15. The problem is then to find the speed at $t = 45$ assuming a linear regression model defined by the data points for $0 \leq t \leq 6$. This means we need to find the scalar parameters a and b of the regression line $s = at_i + b$ where t_i denotes the second and s the speed.

We can use the same strategy as in question 1 of this exercise and compute

$a = \frac{n\sum_{i=1}^{n} t_i s_i - \sum_{i=1}^{n} t_i \sum_{i=1}^{n} s_i}{n\sum_{i=1}^{n} t_i^2 - \left(\sum_{i=1}^{n} t_i\right)^2}$

$b = \frac{1}{n}\left(\sum_{i=1}^{n} s_i - a\sum_{i=1}^{n} t_i\right)$

From Table 4, we can compute,

$n = 7$

$\sum_{i=1}^{n} t_i s_i = 2674$

$\sum_{i=1}^{n} t_i = 21$

$\sum_{i=1}^{n} s_i = 867$

$\sum_{i=1}^{n} t_i^2 = 91$

which result in,

$a = (7 * 2674 - 21 * 867)/(7 * 91 - 21^2) = 2,607$

$b = (1/7) * (867 - 2,607 * 21) = 116,036$

Finally, the speed at $t = 45$ is given by,

$s = 2,607 * 45 + 116,036 = 233,35$

Homework 3 Solution: Computer Vision and Image Analysis
3 Parts, 3 Pages

Interdisciplinary Space Master (ISM)/Master in Information and Computer Science (MICS)
Interdisciplinary Centre for Security, Reliability and Trust
Computer Vision, Imaging, and Machine Intelligence (CVI^2) group

Course Responsible: Prof. Djamila Aouada
TA: Dr. Oyebade K. Oyedotun
djamila.aouada@uni.lu; oyebade.oyedotun@uni.lu

14 October 2020
Due date: 28 October 2020 at 13:00

Instruction for homework

Part I and Part II are compulsory questions. Part III is optional, but carries bonus points.

Part I: Multilayer perceptron (MLP) network computation

1. Construct a multilayer perceptron (MLP) network with four input units, three hidden units and two output units. Initialize all the weights (parameters) of the MLP network to some random values in the range -1 to +1. Let the inputs of the MLP network be 0.4, 0.1, 0.9 and 0.3.

   (a) Given that the units in the hidden and output layers use rectified linear and linear activation functions, respectively, calculate the final outputs of the MLP network.
   Ans: Result is subject to the chosen weights' values.

   (b) Given that the units in the hidden and output layers both use log-sigmoid activation function, calculate the final outputs of the MLP network. (Slides 14, 15 & 20).
   Ans: Result is subject to the chosen weights' values.

(c) Given that the units in the hidden and output layers use rectified linear and softmax activation functions, respectively, calculate the final outputs of the MLP network. (Slides 14, 15 & 20).
Ans: Result is subject to the chosen weights' values.

(d) What are the dimensions of the weight matrices in the hidden and output layers. (Slides 14, 15 & 20).
Ans: Hidden layer weight matrix has a dimension of 4×3. The output layer weight matrix has a dimension of 3×2. (Slides 14, 15 & 20)

2. Assuming you are given a dataset with images of size 16×16 pixels, where each image belongs to one of the five different classes (categories) in the dataset. Construct a MLP network illustrated by a figure (diagram) with two hidden layers that maps the input image to the output classes. Particularly, show clearly your choice for
(a) The number of units in the hidden and output layers.
Ans: The input layer has 256 units. The number of units in the hidden layer is a hyperparameter, which is greater than zero. The number of units in the output layer is five. (Slides 20).
(b) Activation functions for the hidden and output layers.
Ans: Activation functions for the hidden layer is any differentiable non-linear function. The activation function in the output layer is the softmax function. (Slides 15).

$$I = \begin{bmatrix} 0.3445 & 0.9231 & 0.2645 & 0.8753 & 0.3549 & 0.1210 \\ 0.8583 & 0.4956 & 0.7730 & 0.3591 & 0.0128 & 0.9014 \\ 0.0583 & 0.7343 & 0.0289 & 0.5455 & 0.7281 & 0.2451 \\ 0.9533 & 0.0031 & 0.3591 & 0.2743 & 0.8001 & 0.6710 \\ 0.2704 & 0.8451 & 0.0144 & 0.7451 & 0.6423 & 0.3493 \\ 0.5649 & 0.2385 & 0.9971 & 0.0518 & 0.1473 & 0.0089 \end{bmatrix}$$

$$K = \begin{bmatrix} -0.3431 & 0.0012 & 0.8126 \\ 0.8128 & 0.7421 & -0.0513 \\ 0.4789 & 0.2981 & 0.6827 \end{bmatrix}$$

Part II: Convolutional neural network (CNN) computation

1. Given an image of size 28×28 pixels, what is the dimension of the output of the convolution operation using:
(a) A filter of size 5×5, considering that the convolution operation is without padding, and uses a stride of 1.
Ans: Dimension of output is 24×24. (Slides 26).
(b) A filter of size 4×4, considering that the convolution operation is

without padding, and uses a stride of 2.
Ans: Dimension of output is 13×13. (Slides 26).
(c) A filter of size 5×5, considering that the convolution operation is with zero padding, and uses a stride of 1.
Ans: Dimension of output is 26×26. (Slides 26).

2. Given the image and filter shown as I and K, respectively, compute and report as matrices the outputs of the following operations.
(a) A convolution operation without padding that uses a stride of 1.
Ans:

$$Conv\ output = \begin{bmatrix} 1.6274 & 2.3779 & 0.6230 & 1.8451 \\ 2.3208 & 1.0767 & 1.6537 & 1.3858 \\ 0.7336 & 1.6915 & 1.1587 & 2.3183 \\ 1.5659 & 0.9907 & 2.5441 & 1.5087 \end{bmatrix}$$

(Slides 24).

$$Corr\ output = \begin{bmatrix} 1.390 & 2.086 & 1.766 & 0.699 \\ 1.628 & 1.009 & 0.937 & 2.408 \\ 1.154 & 1.364 & 1.704 & 1.582 \\ 1.833 & 1.328 & 1.652 & 1.591 \end{bmatrix}$$

(Slides 24).

(b) A convolution operation without padding that uses a stride of 2.
Ans:

$$Conv\ output = \begin{bmatrix} 1.627 & 0.623 \\ 0.734 & 1.159 \end{bmatrix}$$

(Slides 24).

$$Corr\ output = \begin{bmatrix} 1.390 & 1.766 \\ 1.154 & 1.704 \end{bmatrix}$$

(Slides 24).

(c) A convolution operation with zero padding that uses a stride of 2.
Ans:

$$Conv\ output = \begin{bmatrix} 0.837 & 1.141 & 0.299 \\ 1.133 & 1.077 & 1.386 \\ 1.092 & 0.991 & 1.509 \end{bmatrix}$$

(Slides 24).

$$Corr\ output = \begin{bmatrix} 0.803 & 1.615 & 1.760 \\ 0.696 & 1.009 & 2.408 \\ 0.492 & 1.328 & 1.591 \end{bmatrix}$$

(Slides 24).

3. Given the image I, compute and report as matrices the following.
   (a) The output of applying a max-pooling operation with window size 3 $\times$ 3 pixels.
   Ans:
   $$Max - pooling\ output = \begin{bmatrix} 0.9231 & 0.9014 \\ 0.9971 & 0.8001 \end{bmatrix}$$
   (Slides 30).

   (b) The output of applying an average-pooling operation with window size 3 $\times$ 3 pixels.
   Ans:
   $$Average - pooling\ output = \begin{bmatrix} 0.4978 & 0.4604 \\ 0.4718 & 0.4100 \end{bmatrix}$$
   (Slides 30).

   (c) The output of applying a max-pooling operation with window size 2 $\times$ 2 pixels.
   Ans:
   $$Max - pooling\ outputt = \begin{bmatrix} 0.9231 & 0.8753 & 0.9014 \\ 0.9533 & 0.5455 & 0.8001 \\ 0.8451 & 0.9971 & 0.6423 \end{bmatrix}$$
   (Slides 30).

   (d) The output of applying a global max-pooling operation.
   Ans: 0.9971. (Slides 31).
   (e) The output of applying a global average-pooling operation.
   Ans: 0.4600. (Slides 31).

Part III (Optional): General questions

1. Briefly discuss the usefulness of the activation functions in neural networks. Answer should be limited to a paragraph.
   Ans: The activation function in a neural network introduces non-linearities that empower the model to learn or capture non-linear statistical patterns or relationships in the training data.

2. Briefly (3-5 sentences) discuss why the inputs of neural networks are typically in the range 0 to 1 or -1 to +1.
   Ans: Normalizing the inputs of neural networks in the range 0 to 1 or -1 to +1 reduces the dominance of features with large values on other features with small values during training. Furthermore, normalizing the inputs positions most of the outputs of the hidden units outside of the saturation regime of most activation functions. As such, optimization problems that ensue from the saturation of units' outputs can be circumnaviagted.

3. What are hyperparameters for neural network training (optimization)? Give at least three examples of hyperparameters for neural networks. Answer should be limited to a paragraph.
   Ans: Neural network hyperparameters are training parameters that are determined heuristically during optimization. Different hyperparameters relate to different models and datasets. Examples of hyperparameters include learning rate, momentum rate, batch size, number of units or filters in the different layers, activation function, weight decay magnitude for regularizing model weights, dropout ratios for regularizing units in the different hidden layers, number of training epochs or iterations.

Homework 4: Computer Vision and Image Analysis
2 Parts, 2 Pages

Interdisciplinary Space Master (ISM)/Master in Information and Computer Science (MICS)
Interdisciplinary Centre for Security, Reliability and Trust
Computer Vision, Imaging, and Machine Intelligence (CVI^2) group

Course Responsible: Prof. Djamila Aouada
Dr. Abd El Rahman Shabayek
djamila.aouada@uni.lu; abdelrahman.shabayek@uni.lu

19 October 2020
Due date: 04 November 2020 at 13:00

Part I: Camera Models

1. Given a 3D point P = (30,40,5):
a) what are the normalized image coordinates of p?
$\hat{x} = d * X/Z = 1 * 30/5 = 6$
$\hat{y} = d * Y/Z = 1 * 40/5 = 8$
b) what are the image plane coordinates of p where, the sensor has a zero-skew, a unit aspect-ratio, $f = 2$, $x_0 = 512$ and $y_0 = 512$?
$x = f * \hat{x} + x_0 = 2 * 6 + 512 = 524$
$y = f * \hat{y} + y_0 = 2 * 8 + 512 = 528$
2. An arbitrary combination of a curved mirror and a perspective lens would be:
a) central
b) non-central
c) none of the above

Part II: Multiview Geometry

1. The epipolar plane is defined by:
a) two parallel rays connecting the origin of two different cameras

b) two intersecting rays connecting the origin of two different cameras with a single 3D point
c) none of the above

2. True or false: The essential matrix of two uncalibrated cameras can be directly estimated from multiple correspondences.

3. The epipole is the projection of the:
a) optical center of one camera on the image plane of another camera
b) optical center of one camera on its image plane
c) 3D point onto the epipolar plane
d) none of the above

4. The epipolar constraint maps:
a) a point to its epipolar line in a second image
b) a line in one image to an epipolar line in a second image
c) none of the above

5. True or false: Essential matrices are singular and their rank is 3.

6. True or false: The fundamental matrix can be expressed in terms of both the intrinsic and extrinsic parameters.

7. Given a set of corresponding 2D/3D points in two or more images, computing the extrinsic camera parameters means to estimate:
a) Camera Motion
b) Stereo correspondence
c) Structure from motion
d) Optical flow
e) none of the above

8. True or false:
a) In Euclidean geometry, two lines always intersect. False
b) A vanishing point can be computed in orthographic views. False
c) A vanishing point can lie outside the perspective image plane. True

9. In projective space, a point can be found by computing the:
a) cross product of its intersecting lines.
b) dot product of its intersecting lines.
c) cross product of two parallel lines.
d) dot product of two parallel lines.
e) none of the above.

Homework 5: Computer Vision and Image Analysis - Solutions
2 Parts, 2 Pages

Interdisciplinary Space Master (ISM)/Master in Information and Computer Science (MICS)
Interdisciplinary Centre for Security, Reliability and Trust
Computer Vision, Imaging, and Machine Intelligence (CVI^2) group

Course Responsible: Prof. Djamila Aouada
TA: Dr. Kassem Al Ismaeil
djamila.aouada@uni.lu; kassem.alismaeil@uni.lu

27 October 2020

Deadline: November 4, 2020

1 Part I

1. In the following question, please choose the correct option:
   In the context of the Structure from Motion problem solving, the estimated projection matrices in the case of calibrated cameras:

   (a) preserve parallelism, volume ratios of the reconstructed shape
   (b) preserve angles, ratios of length of the reconstructed shape
   (c) preserve intersection and tangency of the reconstructed shape

   Answer:(b)

2. In the following question, please choose the correct option:
   The fundamental matrix has:

   (a) 5 DoF
   (b) 6 DoF
   (c) 7 DoF
   (d) 8 Dof

   Answer:(c)

3. True or false: The rank of the measurement matrix in the affine structure from motion corresponds to the smallest dimension of the matrices that are forming it.

   Answer:True

4. True or false: The rank of a 3×3 fundamental matrix is at max 3.
   Answer: False, The rank of the fundamental matrix is 2.

2 Part II

1. Given the image below, calculate the image gradients for the highlighted 3×3 patch.

1	1	1	1	1
1	7	8	8	1
1	5	6	9	1
1	8	2	3	3
1	8	2	3	4

Answer: $I_x = \begin{bmatrix} - & 0 & 0 & 0 & - \\ - & 7 & 1 & -7 & - \\ - & 5 & 4 & -5 & - \\ - & 1 & -5 & 1 & - \\ - & 1 & -5 & 2 & - \end{bmatrix}$

Answer: $I_y = \begin{bmatrix} - & - & - & - & - \\ 0 & 4 & 5 & 8 & 0 \\ 0 & 1 & -6 & -5 & 2 \\ 0 & 3 & -4 & -6 & 3 \\ - & - & - & - & - \end{bmatrix}$

2. In the following question, please choose the correct option:
   In the Lucas-Kanade equation, let λ_1 and λ_2 denote the eigenvalues of the matrix $A^T A$. The matrix $A^T A$ is well conditioned if:

   (a) λ_1 and λ_2 are small
   (b) λ_1 and λ_2 are lagre
   (c) λ_1 is large and λ_2 is small
   (d) λ_1 and λ_2 are too large

   Answer:(b)

3. In the following question, please choose the correct option:
   While estimating the optical flow between two consecutive frames, if the motion is small (pixels only move a little bit):

   (a) we can do pixel to pixel comparison

   (b) we can apply a first order approximation on the brightness constancy constraint

   (c) we have to apply a coarse-to-fine optical flow

   Answer:(b)

4. In the following question, please choose the correct option (more than one option can be correct):
   Coarse-to-fine optical flow algorithm :

   (a) is an algorithm to support a large motion estimation between two consecutive frames

   (b) has no direct impact on the accuracy of the estimated motion

   (c) ensures that the small motion assumption remains valid

   Answer:(a) and (c)

Homework 6: Computer Vision and Image Analysis - Solutions
2 Parts, 2 Pages

Interdisciplinary Space Master (ISM)/Master in Information and Computer Science (MICS)
Interdisciplinary Centre for Security, Reliability and Trust
Computer Vision, Imaging, and Machine Intelligence (CVI^2) group

Course Responsible: Prof. Djamila Aouada
TA: Dr. Enjie Ghorbel
djamila.aouada@uni.lu; enjie.ghorbel@uni.lu

November 11, 2020

Let $\mathcal{O}$ be a 3D object and $\mathcal{C}$ a camera fully observing the object $\mathcal{O}$. Let $\mathcal{I}$ be the 2D image captured by $\mathcal{C}$, showing, therefore, the 2D projection of $\mathcal{O}$.

Part I

1. For defining the spatial relationship between $\mathcal{O}$ and $\mathcal{C}$, is the Euclidean distance between the gravity centers of $\mathcal{O}$ and $\mathcal{C}$ sufficient? Explain why (illustrations can be used).

   Answer: The translation between the centers of gravity is not sufficient for defining the spatial relationship between $\mathcal{O}$ and $\mathcal{C}$. Indeed, the attitude of $\mathcal{O}$ with respect to $\mathcal{C}$ can vary while conserving the same translation; thus, resulting on a different spatial distribution. The figure shown in Slide 6 of Lecture 7 can be used for illustrating this.

2. Propose a mathematical representation for modeling this spatial relationship in a single entity.

   Answer: For defining the spatial relationship between $\mathcal{O}$ and $\mathcal{C}$, two main elements are needed: 1) translation and 2) attitude. To model these two elements in a single mathematical entity, a Euclidean transformation matrix which combines a rotation matrix (defining the attitude) and a translation vector can be used (see Slide 14 of Lecture 7).

3. To which mathematical group does this representation belongs to? Define this set mathematically.

   Answer: Euclidean transformation matrices belong to the Special Euclidean group usually denoted by $\mathbb{SE}(3)$.

   $\mathbb{SE}(3) = \{\mathtt{T} \in \mathbb{R}^{4\times4}$ such that, $\mathtt{T} = \begin{bmatrix} \mathtt{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}$ with $\mathtt{R} \in \mathbb{SO}(3)$ and $\mathbf{t} \in \mathbb{R}^3$ $\}$

   In turn, $\mathbb{SO}(3)$ represents the Special Orthogonal group and is composed of rotation matrices. It can be defined as follows,

   $\mathbb{SO}(3) = \{\mathtt{R} \in \mathbb{R}^{3\times3}$ such that, $\mathtt{R}.\mathtt{R}^\mathtt{T} = 1$ and $det(\mathtt{R}) = 1$ $\}$

4. What is the number of parameters defining this representation? Explain.

   Answer: This representation is defined in total by 6 parameters. Indeed, the rotation matrix mainly depends on the three angles of rotations around the axes (X, Y, Z) while the translation depends on the 3 displacements along (X, Y, Z).

Part II

Let us assume that the pose between $\mathcal{C}$ and $\mathcal{O}$ is defined by the following transformation matrix $\mathtt{T} \in \mathbb{SE}(3)$,

$$\mathtt{T} = \begin{bmatrix} cos(\theta) & 0 & sin(\theta) & x_A \\ 0 & 1 & 0 & 0 \\ -sin(\theta) & 0 & cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

with $\theta \in [0, 2\pi]$ and $x_A \in \mathbb{R}$.

1. Decompose the matrix $\mathtt{T}$ to a rotation matrix $\mathtt{R} \in \mathbb{SO}(3)$ and a translation vector $\mathbf{t} \in \mathbb{R}^3$.

   Answer: The matrix T is decomposed as follows: $\mathtt{R} = \begin{bmatrix} cos(\theta) & 0 & sin(\theta) \\ 0 & 1 & 0 \\ -sin(\theta) & 0 & cos(\theta) \end{bmatrix}$ and $\mathbf{t} = \begin{bmatrix} x_A \\ 0 \\ 0 \end{bmatrix}$.

2. Check that the matrix $\mathtt{R} \in \mathbb{SO}(3)$.

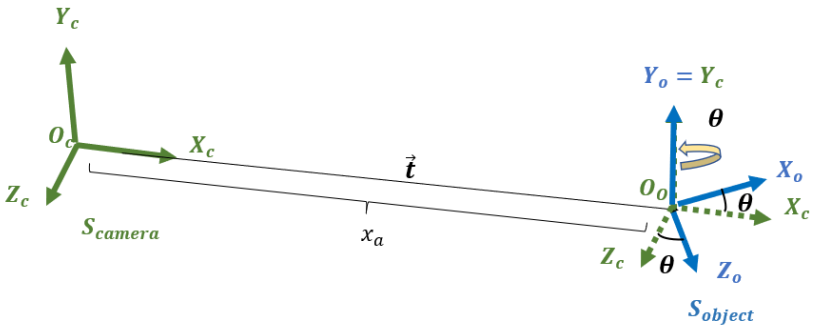
   Answer: Following the definition of $\mathbb{SO}(3)$ (see answer to question 3, Part I), $\mathtt{R} \in \mathbb{SO}(3) \iff \mathrm{R}.\mathrm{R}^T = \mathtt{I}$ and $det(\mathtt{R}) = 1$. Then, we compute the determinant of $\mathtt{R}$ as well as $\mathtt{R}.\mathtt{R}^T$.

$$det(\mathtt{R}) = \begin{vmatrix} cos(\theta) & 0 & sin(\theta) \\ 0 & 1 & 0 \\ -sin(\theta) & 0 & cos(\theta) \end{vmatrix} = cos(\theta) \begin{vmatrix} 1 & 0 \\ 0 & cos(\theta) \end{vmatrix} - 0 \begin{vmatrix} 0 & 0 \\ -sin(\theta) & cos(\theta) \end{vmatrix} +$$
$$sin(\theta) \begin{vmatrix} 0 & 1 \\ -sin(\theta) & 0 \end{vmatrix} = cos(\theta)(cos(\theta) - 0) + sin(\theta)(0 - (-sin(\theta)) = cos^2(\theta) +$$
$$sin^2(\theta) = 1$$
$$\mathtt{R}\mathtt{R}^\mathtt{T} = \begin{bmatrix} cos(\theta) & 0 & sin(\theta) \\ 0 & 1 & 0 \\ -sin(\theta) & 0 & cos(\theta) \end{bmatrix} \begin{bmatrix} cos(\theta) & 0 & -sin(\theta) \\ 0 & 1 & 0 \\ sin(\theta) & 0 & cos(\theta) \end{bmatrix} =$$
$$\begin{bmatrix} cos^2(\theta) + sin^2(\theta) & 0 & sin(\theta)cos(\theta) - sin(\theta)cos(\theta) \\ 0 & 1 & 0 \\ cos(\theta)sin(\theta) - cos(\theta)sin(\theta) & 0 & sin^2(\theta) + cos^2(\theta) \end{bmatrix} =$$
$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \mathtt{I}$$

Since we check that $\mathtt{R}\mathtt{R}^\mathtt{T} = \mathtt{I}$ and $det(\mathtt{R}) = 1$, we check that $\mathtt{R} \in \mathbb{SO}(3)$

3. Given the rotation matrix $\mathtt{R}$ and the translation vector t, illustrate on a graph the object and the camera coordinate systems and annotate all the axes.

   Answer:



Explanation of the figure: Let us define by $S_{object} = (O_o, \mathbf{X}_o, \mathbf{Y}_o, \mathbf{Z_o})$ and $S_{camera} = (O_c, \mathbf{X}_c, \mathbf{Y}_c, \mathbf{Z_c})$ the coordinate systems of the object $\mathcal{O}$ and the camera $\mathcal{C}$, respectively. Let us assume that $\mathbf{X}_o = \overrightarrow{O_oA_o}$, $\mathbf{Y}_o = \overrightarrow{O_oB_o}$, $\mathbf{Z}_o = \overrightarrow{O_oC_o}$, $\mathbf{X}_c = \overrightarrow{O_cA_c}$, $\mathbf{Y}_c = \overrightarrow{O_cB_c}$, $\mathbf{Z}_c = \overrightarrow{O_cC_c}$.

Given the transformation matrix $\mathtt{T}$, we can determine the relations between S_{camera} and S_{object} by expressing O_o, A_o, B_o, C_o in the S_{camera} coordinate system as follows,

$$\begin{bmatrix} O_o \\ 1 \end{bmatrix} = \mathtt{T} \begin{bmatrix} O_c \\ 1 \end{bmatrix} = \mathtt{T} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} x_A \\ 0 \\ 0 \\ 1 \end{bmatrix}, \text{ then } O_o = (x_A, 0, 0)_{S_{camera}}$$

$$\begin{bmatrix} A_o \\ 1 \end{bmatrix} = \mathtt{T} \begin{bmatrix} A_c \\ 1 \end{bmatrix} = \mathtt{T} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} cos(\theta) + x_A \\ 0 \\ -sin(\theta) \\ 1 \end{bmatrix}, \text{ then } A_o = (cos(\theta) + x_A, 0, sin(\theta))_{S_{camera}}$$

$$\begin{bmatrix} B_o \\ 1 \end{bmatrix} = \mathtt{T} \begin{bmatrix} B_c \\ 1 \end{bmatrix} = \mathtt{T} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} cos(\theta) + x_A \\ 0 \\ -sin(\theta) \\ 1 \end{bmatrix}, \text{ then } B_o = (x_A, 1, 0)_{S_{camera}}$$

$$\begin{bmatrix} C_o \\ 1 \end{bmatrix} = \mathtt{T} \begin{bmatrix} C_c \\ 1 \end{bmatrix} = \mathtt{T} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} sin(\theta) + x_A \\ 0 \\ cos(\theta) \\ 1 \end{bmatrix}, \text{ then } C_o = (sin(\theta) + x_A, 0, cos(\theta))_{S_{camera}}$$

Then, $\mathbf{X}_o = A_o - O_o = \begin{bmatrix} cos(\theta) + x_A - x_A \\ 0 - 0 \\ -sin(\theta) \end{bmatrix} = \begin{bmatrix} cos(\theta) \\ 0 \\ -sin(\theta) \end{bmatrix}$

Then, $\mathbf{Y}_o = B_o - O_o = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$

Then, $\mathbf{Z}_o = C_o - O_o = \begin{bmatrix} sin(\theta) \\ 0 \\ cos(\theta) \end{bmatrix}$

Thus, only one rotation around the axis Y by an angle θ can be noted. To determine the direction of rotation, we need to check the signs. According to the signs, we can see that it is a clockwise rotation going from (Z_c to X_c).

The translation is expressed with respect to only one component (the direction $\mathbf{X}_c$). In the exercise, we did not specify if x_A is positive or not. So, we can consider it as positive or negative when drawing the illustration (following the positive or negative directions of $\mathbf{X}_c$). Both answers are correct. In the figure shown here, we assume that $x_A > 0$.

4. Let us define the intrinsic camera parameter matrix as

$$\mathtt{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

(a) Explain the role of the matrix K. Explain what are the physical meaning of the values f_x, f_y, c_x and c_y?

Answer: The intrinsic parameter camera matrix maps 3D camera coordinates to 3D pixel image coordinates. f_x and f_y represent the focal lengths of the camera which convert the distances (m,mm) in pixels with respect to the axes $\mathbf{X}$ and $\mathbf{Y}$. c_x and c_y represent respectively the x- and y-offsets of the image coordinate system with respect to the camera coordinate system.

(b) Compute the projection matrix P defining the relationship between each point $\mathbf{w} = (u, v, 1)$ in the image plane $\mathcal{I}$ and its corresponding homogeneous 3D coordinates $\mathbf{X} = (x, y, z, 1)$ in the coordinate system of $\mathcal{O}$.

$$\mathtt{P} = \mathtt{K}.\widehat{\mathtt{T}} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos(\theta) & 0 & sin(\theta) & x_A \\ 0 & 1 & 0 & 0 \\ -sin(\theta) & 0 & cos(\theta) & 0 \end{bmatrix} =$$
$$\begin{bmatrix} f_x cos(\theta) - c_x sin(\theta) & 0 & f_x sin(\theta) + c_x cos(\theta) & f_x x_A \\ -c_y sin(\theta) & f_y & c_y cos(\theta) & 0 \\ -sin(\theta) & 0 & cos(\theta) & 0 \end{bmatrix}$$

5. What is the RQ factorization?

Note: You will probably find more materials in books and Internet about the QR factorization. But, here we are interested in the RQ factorization.

Answer: The RQ factorization allows the factorisation of an $n \times n$ matrix `A` using an orthogonal matrix `Q` and a right triangular matrix `R` such that `A = RQ` (see Slide 34 of Lecture 7).

6. What is the interest of using the RQ estimation for estimating the pose?

Answer: As the pose is composed of a rotation matrix that is orthogonal and that the intrinsic camera matrix is right triangular, if we know the full projection matrix, we can determine the pose using this factorization.

7. The Direct Linear Transformations (DLT) algorithm is widely used for determining the pose of an object using a single image. Let assume $(\mathbf{X_i}, \mathbf{w_i})_{i \in \{1,...N\}}$ to be the respective correspondences between the 3D object such that $\mathbf{X}_i \in \mathbb{R}^3$ and the projected object in $\mathcal{I}$ such that $\mathbf{w}_i \in \mathbb{R}^2$.

(a) What is the minimum number of correspondences needed to determine the pose? Explain why.

Answer: The DLT algorithm starts by formulating the following correspondences using N equalities resulting from the pinhole approximation as follows,

$\lambda_i \mathbf{w}_i = \mathtt{P}\mathbf{X}_i$ with $i \in \{1, ..., N\}$, $\mathtt{P}$ the camera projection matrix and $\lambda_i \in \mathbb{R}$.

Each equality will gives N equations. Then, in total, we will have $3N$ equations. Since we have N unknowns λ_i and $\mathtt{P}$ is formed by 11 unknown parameters (6 parameters for the extrinsic camera matrix and 5 for the intrinsic matrix). Then, in total, we have $11 + N$ unknowns. To resolve this equation system, we need to have a number of equations that is superior or equal to the number of unknowns.

$\Rightarrow 3N \geq 11 + N \Rightarrow N \geq 6$ (see Slide 37)

(b) Explain in detail the steps of the DLT.

Answer: The summary of the steps of DLT are given in Slide 43 of Lecture 2:

1) Finding the 3D-2D Correspondences: Let assume we have a 3D model of the object. We need to find a set of correspondences between the 3D points of the object model and the 2D points shown in the image. Feature matching techniques are usually employed.

2) Writing the correspondence in a matrix form $\mathtt{M}\mathbf{v} = 0$. Thus, we obtain an optimization problem with constraints: $\min_{\|v\|} \|Mv\|^2$ subject to $\|v\|^2 = 1$

(the Details of calculation can be found in slides 37-40)

3) Applying Singular Value Decomposition to $\mathtt{M}$ for finding the solutions $\mathbf{v}$ that are the eigenvectors of $\mathtt{M}^T\mathtt{M}$ (Details in Slide 41)

4) Selecting the logical solutions (solution returning negative depth behind the camera are to be rejected).