

**ARTIFICIAL INTELLIGENCE PROGRAMMING PROJECT**

**Report 3 – Data Collection**

– Hanoi, October 2021 –

**Table of Contents**

[III. Data Collection 3](#_heading=h.30j0zll)

[1. Methodology 3](#_heading=h.1fob9te)

[2. Data description 3](#_heading=h.3znysh7)

[3. Data Exploration](#_heading=h.2et92p0) 6

# III. Data Collection

## 1. Methodology

* The data set was graciously provided by Dr. Yoshua Bengio of the University of Montreal. The images must be grayscale and small in dimension.
* For data augmentation, we add 5 methods: Horizontal Flip, Linear Contrast, Gaussian Blur, Rotation, Scaling. In order to expand the size and help generalize future models.

## 2. Data Description

* We have used the dataset from Kaggle Competition - Facial keypoint detection. There are 15 facial keypoints per image: image like left eye center, right eye center, left eye inner corner, left eye outer corner, right eye inner corner, right eye outer corner, left eyebrow inner end, left eyebrow outer end, right eyebrow inner end, right eyebrow outer end, nose tip, mouth left corner, mouth right corner, mouth center top lip, and mouth center bottom lip.
* In some examples, some of the target keypoint positions are missing (encoded as missing entries in the csv, i.e., with nothing between two commas).
* The input image is given in the last field of the data files, and consists of a list of pixels (ordered by row), as integers in (0,255). The images are 96x96 pixels.
* The data file has 2 .csv file:
  + training.csv: contains 7049 images. Each row contains 15 key points in the form of (x,y) (for example: left\_eye\_center\_x, left\_eye\_center\_y), and image as row-ordered list of pixels.
  + test.csv: list of 1783 test images. Each row contains ImageId and image data as row-ordered list of pixels

## 3. Data Exploration

* Using pandas, create a data frame for the .csv file and extract features for analysis.
* Every feature uses float as data type except for “Image” uses the object data type.
* Since the data contain some missing values this step calculates and visualises those missing values for every feature (Fig.1). Overall, Each feature contains around 60 percent missing values so that we must base on the distribution of each feature to determine the solution for handling missing values.

|  |
| --- |
| Fig.1: Missing data |

* The distribution for each feature (Fig.2)

|  |
| --- |
| Fig.2: Distribution table |

* As expected,the key points follow a normal distribution so the missing values can be replaced with the distribution mean as one of the simplest strategies.
* The heatmap in seaborn library is used for displaying the correlation between features (Fig.3).

|  |
| --- |
| Fig.3: Data correlation. |

* Visualising the input image, we create a numpy array of the pixel values in the image column of our training dataset, then use matplotlib to plot the image from the pixel values (Fig.4). After that, the keypoint will be plotted on the face image (Fig.5).

|  |
| --- |
| Fig.4: Input Image |

|  |
| --- |
| Fig.4: Input Image with keypoints |