**Ministry of Science and Higher Education**

**of the Russian Federation**

**ITMO University**

Faculty of Digital Transformations

Educational program 01.04.02 Applied mathematics and informatics\_\_\_\_

Subject area (major) Big Data and Machine Learning\_\_\_\_

REPORT

on practical training Research Internship

Task topic: Investigation of the influence of environmental factors on the psychoemotional state of a person

Student: Yamoldin Alexander, J4134c

Head of Practice from the trainee’s host organization: Basov Oleg, National Center for Cognitive Development, Senior Researcherhead of the laboratory «Center Information Optical Technologies»

Head of Practice from ITMO University: Alexandra Klimova, Faculty of Digital Transformation, assistant professor

|  |  |
| --- | --- |
|  | Practice completed with grade \_\_\_\_  Commission member signatures:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ *full name*  (signature)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ *full name*  (signature)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ *full name*  (signature)  Date \_\_\_\_\_\_\_\_ |

St. Petersburg  
2022

# Annotation

33 pages, 22 pictures, 3 tables, 21 references

MACHINE LEARNING, CONVOLUTIONAL NEURONNET, OBJECT DETECTION, INDOOR AIR QUALITY, PSYCHO-EMOTIONAL STATE, WELL-BEING

**Object of research**: People psycho-emotional state depends on Indoor Air Quality.

**Target of research:** The aim of the work is to determine the existence of a correlation between the psycho-emotional state of a person and the IAQ index.

**Methodology:** Application of Machine Learning and Artificial Intelligence for Dynamic Carbon Dioxide Level and Indoor Temperature Prediction.

**Results:** In this work have reviewed the theoretical aspects of solving the problems of object detection, selected a model for further study, converted the sensor data to the IAQ index, collected experimental data, selected behavioral markers, selected tools for data markup, and performed initial manual markup of the data

Title

[Annotation 1](#_Toc106546433)

[1. The theoretical part of the computer vision concept 3](#_Toc106546434)

[1.1 Brief theory of convolutional neural networks 3](#_Toc106546435)

[1.2 Metrics 7](#_Toc106546436)

[1.3 Object Detection 10](#_Toc106546437)

[1.4 YOLO model 13](#_Toc106546438)

[1.5 Transfer Learning 15](#_Toc106546439)

[1.6 Object tracker 16](#_Toc106546440)

[1.6.1 The object tracking process 16](#_Toc106546441)

[1.6.2 Algorithm for tracking centers 17](#_Toc106546442)

[1.6.3 Tracking algorithm using the Kalman filter 20](#_Toc106546443)

[2. Experiment and data collection 22](#_Toc106546444)

[2.1 Experiment 22](#_Toc106546445)

[2.2 Data collecting 23](#_Toc106546446)

[2.3 Data labeling 28](#_Toc106546447)

[3. Conclusion 32](#_Toc106546448)

[References 34](#_Toc106546449)

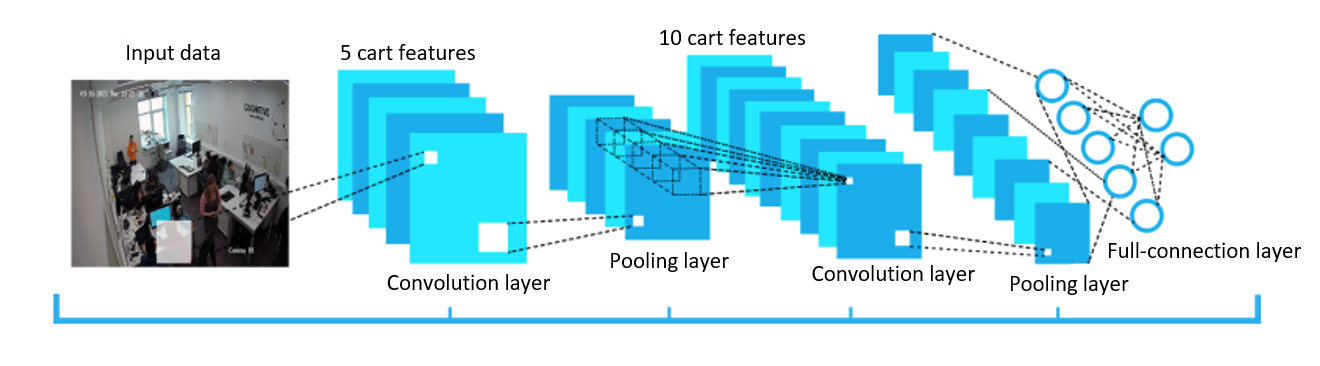
# The theoretical part of the computer vision concept

## Brief theory of convolutional neural networks

The terms 'deep learning' or 'deep neural network' refer to artificial neural networks (ANN). Over the past few decades, they have become one of the most powerful tools for photo/video data analysis. The most popular neural network tool for photo/video data analysis is convolutional neural networks (CNN). The name is taken from a mathematical linear operation between matrices called convolution.

CNNs are a specialized type of neural network model designed to work with 2D image data, although they can be used with one- and three-dimensional data. One of the main problems CNN solves is the recognition of spatially-independent patterns that are resistant to zooming, rotating, changing perspectives and other distortions. For example, in an application for recognizing people, you don't have to pay attention to where they are located in images. The only task is to detect them regardless of their position in a given image. Another important aspect of CNN is to obtain abstract functions while moving into deeper layers. On average, the recognition accuracy of such networks outperforms full-connection neural networks by 10-15%. It also increases the speed of training by paralleling the convolution process for each map, as well as the reverse convolution when the error propagates through the network [1].

In general, a CNN consists of three types of layers: convolutional, pooling and full-connection layers (pic.1) [2].



Picture 1 - Architecture of the convolutional neural network

The input data of each pixel value is normalized by the MinMax scaler to the range [0, 1] as follows

|  |  |
| --- | --- |
|  | (1) |

|  |  |
| --- | --- |
|  | (2) |

where, – data normalization function;

– data variance function;

c – pixel color ϵ [0, 255];

– is the smallest value of the found pixel in the image;

– the highest value of the found pixel in the image.

The basic element of a convolutional neural network is the convolutional layer, which performs an operation called convolution [3]. A convolution is a linear operation that performs multiplication of a set of weights by the input data. The convolution layer consists of cores with additive bias components for each core, and computes the convolution of the output image of the previous layer using each of the cores, adding a bias component each time. The activation function is then applied to the entire output image. Usually, the input stream for a convolutional layer consists of channels. For example, for the input layer in red/green/blue image analysis. The kernels are also expanded so that they consist of channels, so the formula for one channel of the output image looks like this:

|  |  |
| --- | --- |
|  | (3) |

where, K – shift kernel;

I – two-dimensional image;

x – dimensionality of the two-dimensional image along the first axis;

y – dimensionality of the two-dimensional image along the second axis;

– activation function;

b – bias component;

- dimensionality of the K-matrix on the first axis;

- dimensionality of the K-matrix on the second axis;

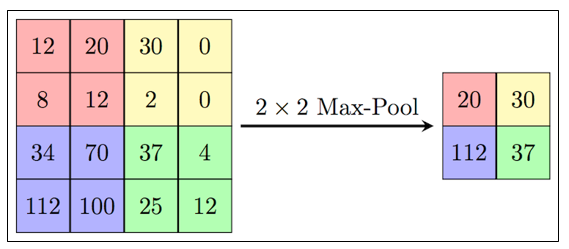
– number of channels;

Using the kernel reduces the computational complexity and aggregates additional information due to the use of weighted sums.

After the convolution layer, it is common to use a pooling layer. This is used to reduce the dependence on accurate positioning of objects by reducing the spatial resolution. Reducing spatial resolution is simply reducing the number of feature map pixels due to the pooling layer [4]. The most popular are two types of pooling:

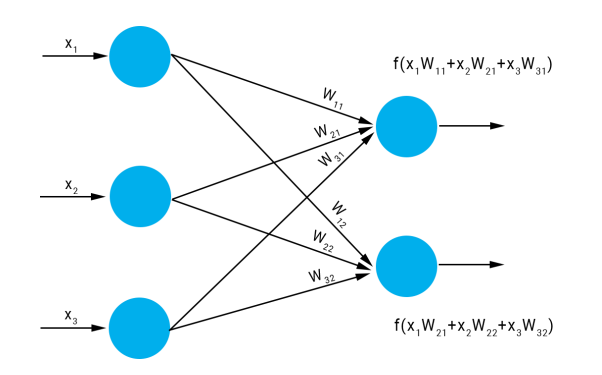
- Average pooling is a variant of pooling, which takes as output data the average value of pixels in the kernel.

- Max pooling (pic. 2) is a version of pooling where the maximum value of pixels in the kernel is taken as output data.



Picture 2 - Forming a new sub-sample layer map based on the previous convolutional layer map. Max Pooling operation [5]

At the end of the convolutional neural network there is a fully connected layer (pic. 3)



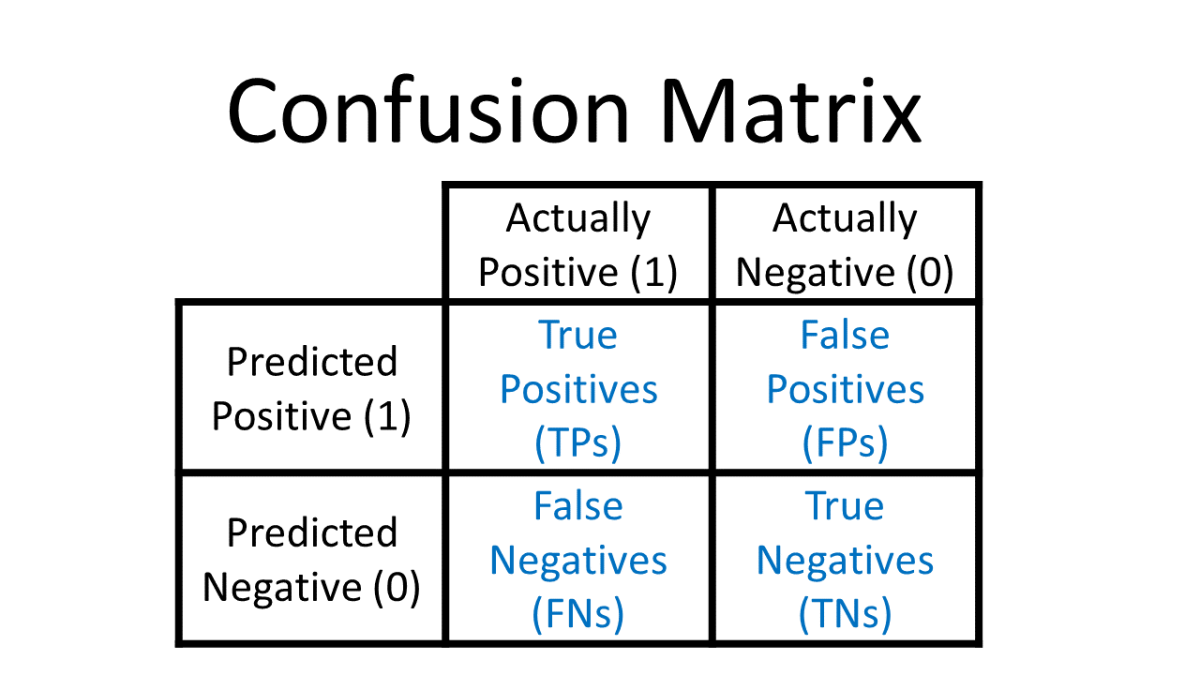
Picture 3 - Fully connected layer

A fully connected layer is a layer in which each neuron receives input from all neurons of its previous layer. In this layer, matrix-vector multiplication is performed, and the values used in the matrix are parameters that can be trained. The result of the full-connection layer is a vector of dimension m at m class classification.

## 1.2 Metrics

MAP (Mean Average Precision) is one of the most common metrics for measuring decoder accuracy. For MAP description we introduce the concept of the following metrics: precision, recall, IoU, as well as the concept of confusion matrix.

Confusion matrix is a diagram of classifier prediction accuracy for two or more classes (pic. 4).



Picture 4 - Confusion matrix [6]

True Positive is the result when the model correctly predicts a positive class.

True Negative is the result when the model correctly predicts a negative class.

False Positive is a result where the model incorrectly predicts a positive class.

False Negative is a result where the model incorrectly predicts a negative class.

Thus, the Precision metric can be introduced as follows:

|  |  |
| --- | --- |
|  | (4) |

where, TP – True Positive;

FP – False Positive.

Precision measures the accuracy of the classifier predictions.

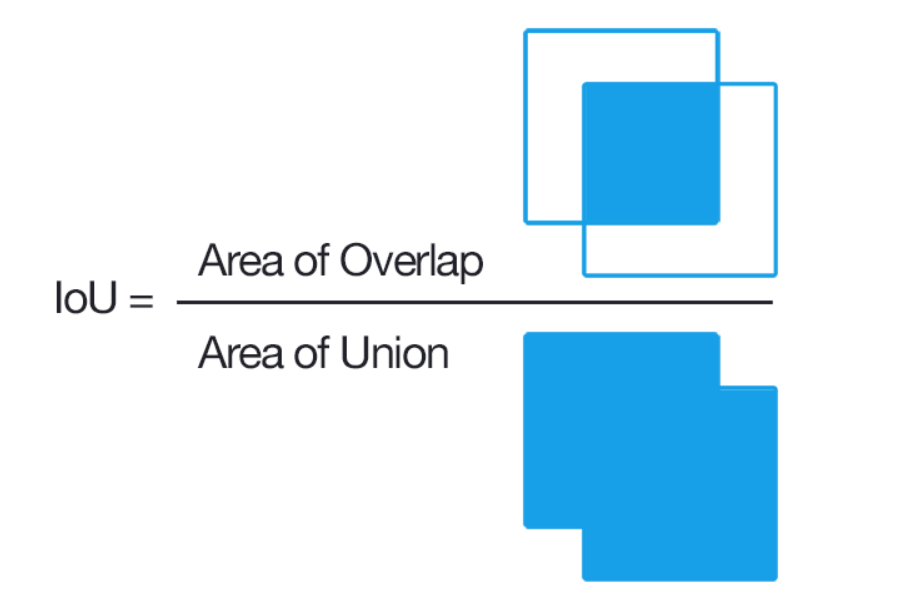
The next metric we introduce is Recall. Recall shows how well the classifier identifies objects of a "positive" class.

|  |  |
| --- | --- |
|  | (5) |

where, TP – True Positive;

FN – False Negative.

The last metric to be introduced is the IoU (Intersection over Union) metric, which takes into account the coverage of the predicted BBox and the ground true BBox. IoU is a number from 0 to 1 indicating how much the two objects have the same internal volume (pic.5).



Picture 5 - Evaluating Intersection over Union

Object detection systems make predictions in terms of the bounding box (BBox) and class label using the IoU value for a given threshold. For example, if the IoU threshold is 0.5 and the IoU value for a forecast is 0.7, then the forecast is classified as true positive (TF). On the other hand, if the IoU is 0.3, then it is classified as false positive (FP).

Average Precision calculates the average precision for Recall in the range from 0 to 1:

|  |  |
| --- | --- |
|  | (6) |

Then Mean Average Precision can be calculated as the average for Average Precision

|  |  |
| --- | --- |
|  | (7) |

where, Q - the number of requests.

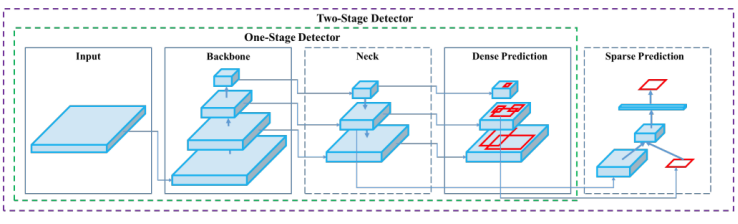
## Object Detection

Object detection is one of the most important tasks of computer vision. Using computer vision it is possible to detect instances of visual objects of a certain class (for example: people, plants, vehicles). A distinctive feature of detection is the detection of the exact coordinates of the objects being searched for as a bounding box (BBox). Object detection is one of the fundamental problems of computer vision, underlying many other tasks, such as: segmentation, safety monitoring and autonomous driving. Most modern detectors use the best deep learning networks as the basis [7].

Current image object detectors can be divided into two categories: One-stage detectors and Two-stage detectors.

Examples of One-Stage Detectors are RetinaNet, YOLO, SSD.

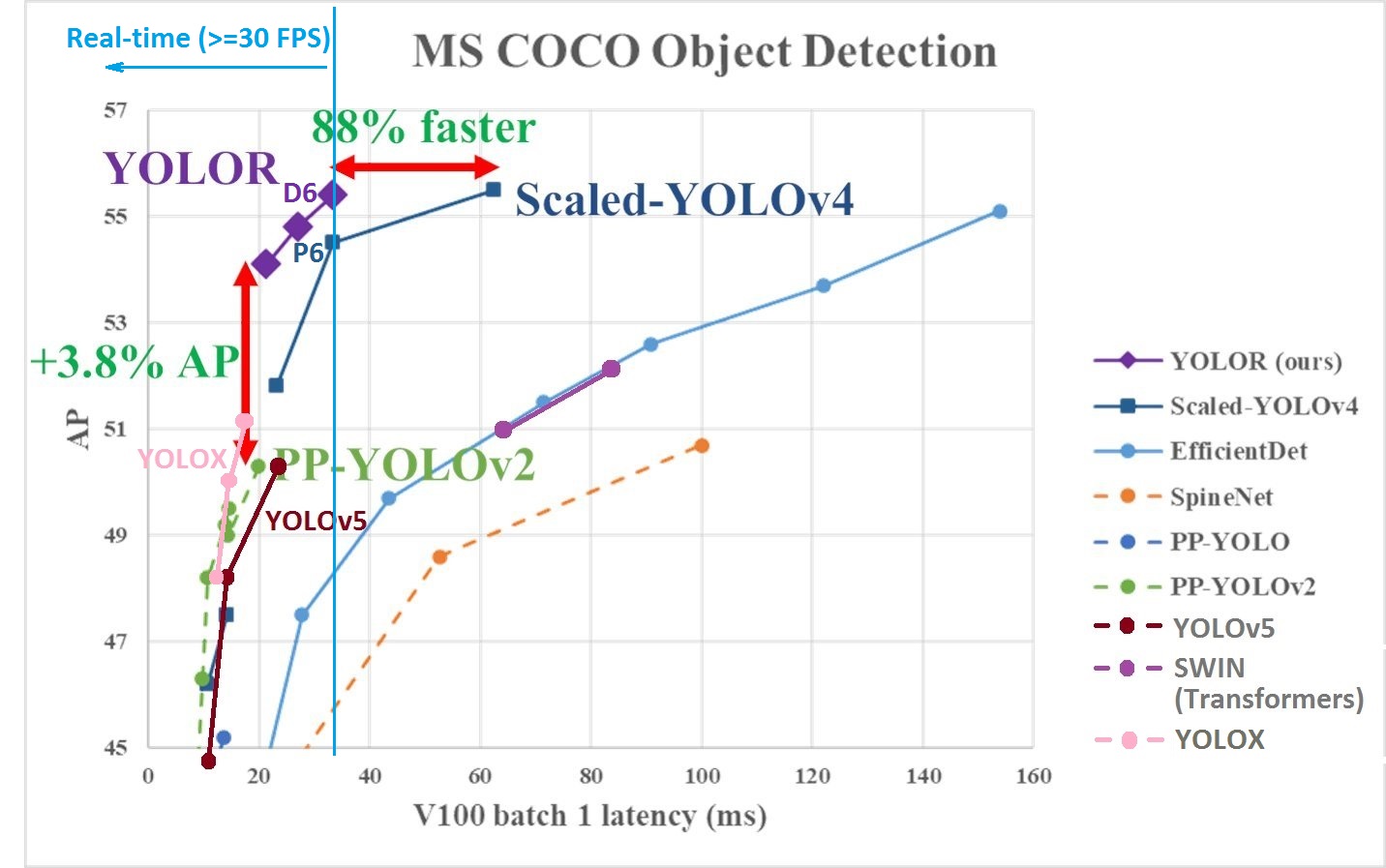
Examples of Two-Stage Detectors are R-CNN, Fast R-CNN, Faster R-CNN.



Picture 6 - The structure of One-Stage and Two-Stage Detectors [8]

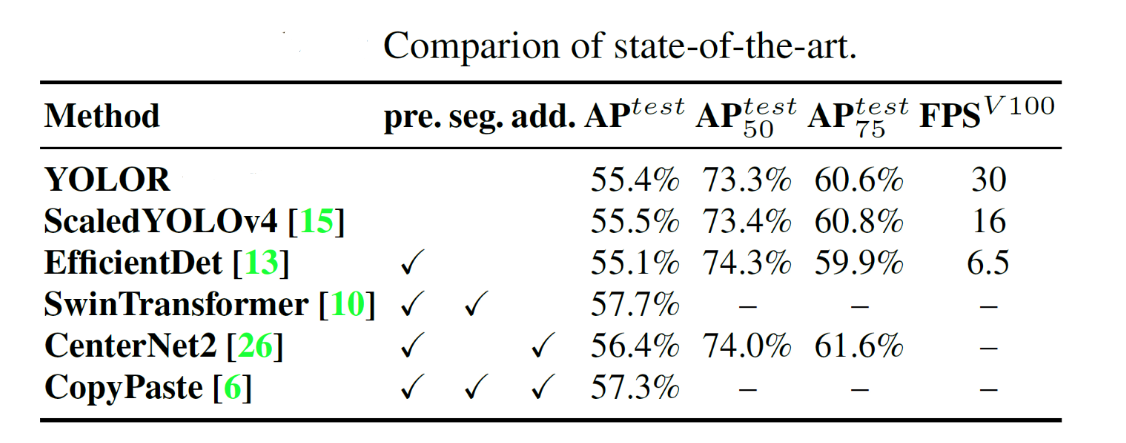
The main difference between the two methods is whether to create a regional suggestion. The authors of the two-stage detectors developed a special network, which uses selective search to extract only 2000 regions from the image. The region proposal network (RPN) is located after the last convolutional layer. These 2000 candidate region proposals are squared and fed into a convolutional neural network, which produces a 4096-dimensional feature vector as output. This network generates suggested regions based on the last convolutional feature map. After this step, a standard scheme is used (Rol pooling of the target region, fully connected layers, and then classification and regression). Two-stage detectors have high accuracy in localizing and recognizing objects, while single-stage detectors provide high output speed. One-stage object detection algorithms do not require the creation of a region proposal. It can directly classify an object and its coordinates. Two-stage object detection algorithms must generate a suggestion area before classifying and positioning them. To directly estimate the coordinate values of each BBox point, we need to treat these points as independent variables. To solve this problem, the researchers suggested using instead of the standard loss function, the IoU loss function.

Object detection models are currently being tested on the Microsoft COCO dataset [9]. This dataset contains 328,000 images with 2.5 million labeled instances of 91 object types that would be easily recognizable by a 4 year old. As of 2022, the best way to detect image objects is to YOLOR [10] and Scaled-YOLOv4 [11] (pic.7).



Picture 7 - Models comparing on MS COCO dataset [10]

This paper will consider the world's best solutions available at the time of writing. Based on the work of [10], it is obvious that the best choice for the problem we are solving is Scaled-YOLOv4 (pic.8).



Picture 8 - Comparison state-of-the-art object detection models

ScaledYOLOv4 loses in FPS, but has the highest recognition accuracy. Since this work analyzes the change of IAQ in the room, which is a highly inertial parameter, we do not need to require the neural network to work directly in real time. Therefore, I gave priority to the quality of object recognition.

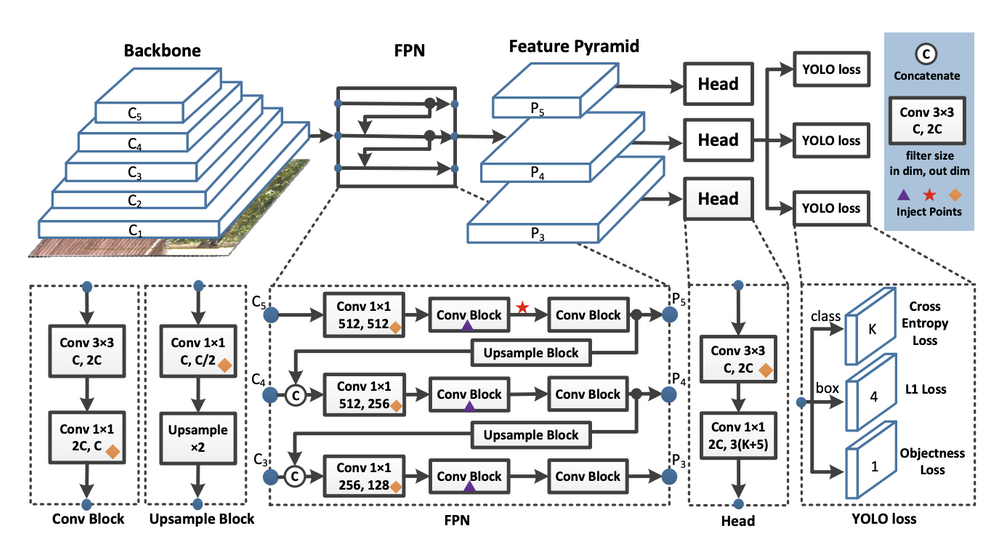
## 1.4 YOLO model

The YOLO (You Only Look Once) model was first published (by Joseph Redmon et al.) in 2015. The original YOLO network uses a modified GoogLeNet as a reference network [12]. Later, a new model called DarkNet-19 was created, which follows the general design of 3 × 3 filters, doubling the number of channels at each merging step. The 1 × 1 filters are also used to periodically compress the representation of an object across the network. YOLOv4 is made on the basis of YOLOv3 with the following changes: YOLOv4 uses CSPDarknet53 convolutional neural network as the backbone, SPP (spatial pyramid pooling) intermediate stage (neck) and (PAN) Path Aggregation Network, YOLOv3 full link final classifier (Dense Prediction). The basic models (Backbone) were pre-trained as image classifiers before being adapted to the detection task. To adapt the classification network into a network for object detection, it is necessary to remove the last few layers of the network and add a convolution layer with filters to produce BBox predictions. A grid is superimposed on the input image, dividing it into S×S regions. For each region, the neural network determines 5 bounding boxes for the object, a confidence score for frame detection, which reflects the degree of confidence of the model that the field contains an object:

|  |  |
| --- | --- |
|  | (8) |

As a result, the network selects the area with the highest level of network response reliability.

Unlike the parent YOLOv4 (pic.9) model the Scaled-YOLOv4 authors apply the concepts laid out in the Cross-Stage Partial Networks [13].



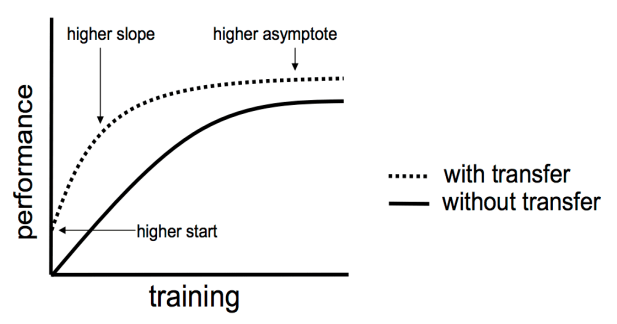
Picture 9 - YOLOv4 architecture

To detect large objects in large images, the authors increase the depth and number of stages in in the CNN backbone and neck. This allows them to first scale up input size and number of stages, and dynamically adjust width and depth according to realtime inference speed requirements.

## 1.5 Transfer Learning

Transfer learning is a machine learning method in which an already trained model is retrained on another related task. Transfer learning allows for much faster progression when learning on a new dataset, which improves the performance of the simulation task many times over. Transfer learning is popular in deep learning, given the huge resources and datasets on which deep learning models are trained. Transfer learning works in deep learning only if the patterns of the original model, are common to the retraining model. First the baseline network is trained on the baseline dataset and task, and then the learned patterns are transferred to a second target network for training on the target dataset and task.

The use of transfer learning has three advantages (pic. 10).



Picture 10 - Transfer learning approach

When using transfer learning we have. A higher start as compared to the original model. due to which the training time is reduced and in parallel the task of initial initialization of weights is discarded. The performance curve has a larger slope tangent, so reaching the asymptotic maximum of the generalization ability of the model is faster. Moreover, the performance asymptote itself lies higher than on the original model. This means that this model achieves a higher convergent skill [14].

## 1.6 Object tracker

Object tracking, is a difficult task. Difficulties with object tracking can arise from abrupt object movement, changes in the appearance patterns of both the object and the scene, overlapping objects, and camera movement. In this section several approaches to object tracking will be investigated, algorithms will be analyzed, pros and cons will be described [15].

### 1.6.1 The object tracking process

- The initial set of detected objects (e.g., the input coordinate set BBox) is taken.

- Creating a unique identifier for each of the initial detections.

- Tracking each of the objects as they move through the frames in the video, keeping the assignment of unique identifiers. The ideal object tracking algorithm should satisfy the following parameters:

- Use the object detection phase only once (that is, when the object is initially detected). - Be quite fast - much faster than starting the object detector itself.

- Be able to handle when a tracked object disappears or goes outside the boundaries of the video frame.

- Be resistant to occlusion.

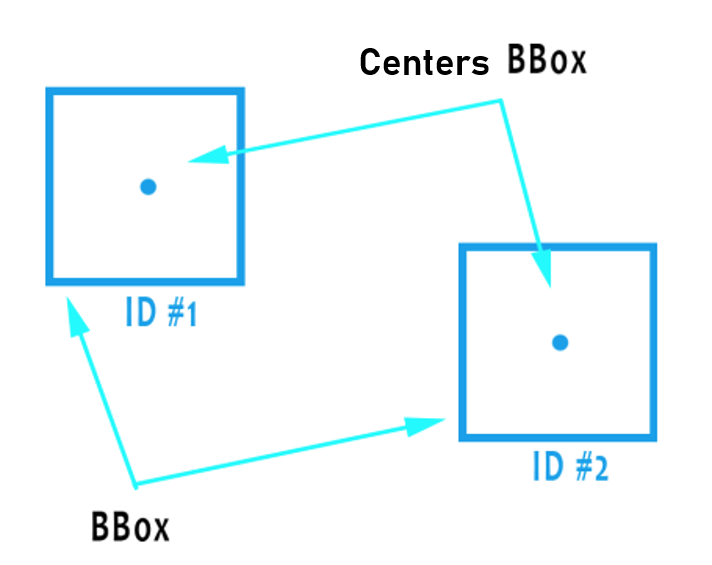
- Be able to pick up objects that have been lost between shots.

### 1.6.2 Algorithm for tracking centers

The object tracking algorithm is called center tracking because it is based on the Euclidean distance between existing object centers and new object centers between successive frames in the video.

The algorithm for tracking centers is a multi-step process:

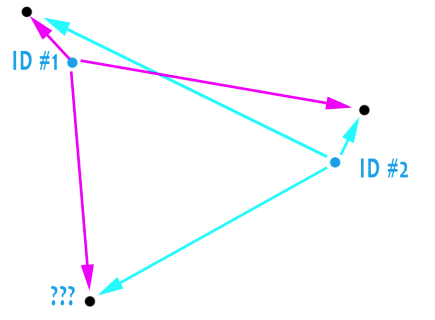
Step 1: take BBox coordinates and calculate its center (pic.11)



Picture 11- Example of centers and body BBox

The center tracking algorithm adopts a set of bounding rectangle coordinates (x, y) for each detected object in each individual frame. Bounding boxes (BBoxes) can be created by any type of object detector (color threshold detection, contour extraction, Haar cascades, HOG, Faster R-CNN, SSD neural networks, YOLO, etc. etc.), as long as they are computed for each frame in the video. Once the detector determines the coordinates of the BBox, you must calculate the coordinates of its center (x, y). Pic. 11 shows taking a set of coordinates of a framing rectangle and calculating the center. Because this is the initial set of BBoxes presented to the algorithm, they are assigned unique identifiers.

Step 2: Calculate the Euclidean distance between new BBoxes and existing objects (pic. 12):



Picture 12. Adding new objects to tracking

For each subsequent frame in the video stream, step 1, calculating object centers, is applied; however, instead of assigning a new unique identifier to each detected object, you must first determine whether the centers of the new objects can be associated with the centers of the old object. To perform this process, a Euclidean distance is calculated for each pair of existing objects and input objects. You can see in pic.12 that three objects are found in the next iteration. The two pairs close to each other are the two existing objects.

Step 3 update (x, y) -coordinates of existing objects:

The algorithm assumes that the object will potentially move between successive frames, but the center distances for frames Ft and Ft + 1 will be smaller than all other distances between objects. Pic.12 shows how the center tracking algorithm selects an association of cents that minimizes their respective Euclidean distances. This leaves one extra point without an association.

Step 4: Register a new object:

If more inputs are detected than existing tracked objects, you must register the new object. Registration means adding a new object to the list of tracked objects and assigning it a new identifier. After that, the algorithm returns to step 2 and repeats the sequence of steps for each frame in the video stream.

Step 5: Unregister old objects:

The object tracking algorithm must be able to handle situations when an object has been lost or has left the field of view. Registration of old objects is cancelled when they cannot be matched with any existing objects for all N subsequent frames.

### 1.6.3 Tracking algorithm using the Kalman filter

This is an advanced version of the tracker, in addition to the distance between the centers of the objects are used their similarity parameters and the speed of the object for prediction. Features of the view outside the detection component are ignored in tracking, only position and size are used to estimate the movement of the bounding rectangle.

Two classic but extremely effective methods, the Kalman filter and the Hungarian algorithm, are used to predict the movement and association of motion data [16]. The interframe movements of each target are approximated using a linear constant velocity model, which is independent of other objects and camera motion. The state of each target is modeled as follows:

|  |  |
| --- | --- |
|  | (9) |

where, - the horizontal location of the center pixels;

- vertical arrangement of the center pixels;

- area of the bounding rectangle;

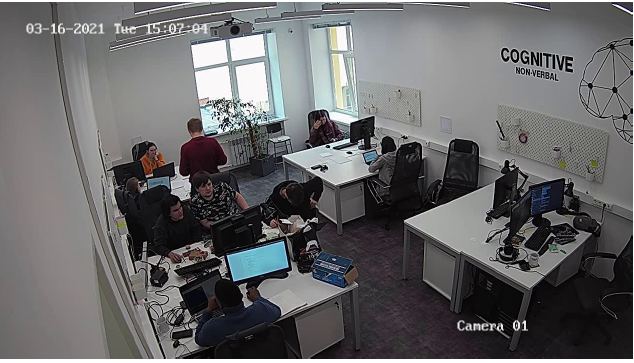
- aspect ratio of the bounding rectangle.

When a detector is triggered, the detected BBox is used to update the tracker status, where speed variables are determined using the Kalman filter framework [17]. When assigning detections to existing targets, the BBox geometry of each target is estimated by predicting its new location in the current frame. The classification cost-matri is calculated as the intersection distance over union (IoU) between each detection and all predicted bounding boxes for existing targets. The IOU distance of the bounding rectangles implicitly takes short-term overlap into account. In particular, when a target is overlapped by another object, only the occlude is detected. When objects enter or leave the scope, unique identifiers must be created or destroyed. To create new objects, any detection with an overlap less than IoUmin is treated as indicating the existence of an untraceable object. The tracker is initialized using a bounding rectangle geometry with zero speed [18]. The new target is then put to the test period, the target must be linked to subsequent detections in order to accumulate enough evidence and avoid tracking down false positives. Tracks are terminated if they are not detected within TLost frames. This prevents the unlimited growth of tracking objects and localisation errors caused by predictions over long periods of time without corrections.

# Experiment and data collection

## 2.1 Experiment

Experimental data were collected in the 408 laboratory of the ITMO National Centre for Cognitive Development. A general view of the laboratory is shown in pic. 13.



Picture 13 - Laboratory 408 of ITMO's National Cognitive Development Centre

This laboratory is equipped with two fixed IP cameras suspended diametrically opposed in the near left and far right upper corner of the classroom, so as not to disturb or annoy the test takers present in the classroom, but at the same time to have a full view of the classroom space and the position of people from two diametrically opposed sides. The lab has a static background and the workstations of the test persons, which makes it easier for the neural network to detect people. In addition, this classroom is equipped with ambient light sensors (## write sensor) and a Bosch BME680 ambient light sensor.

Bosch BME680 is a digital 4-in-1 device capable of measuring air pollution, air humidity, atmospheric pressure and ambient temperature [19]. The Bosch sensor requires pre-calibration to a reference level [19], so the sensor was calibrated to the lab's reference contamination level before starting work.

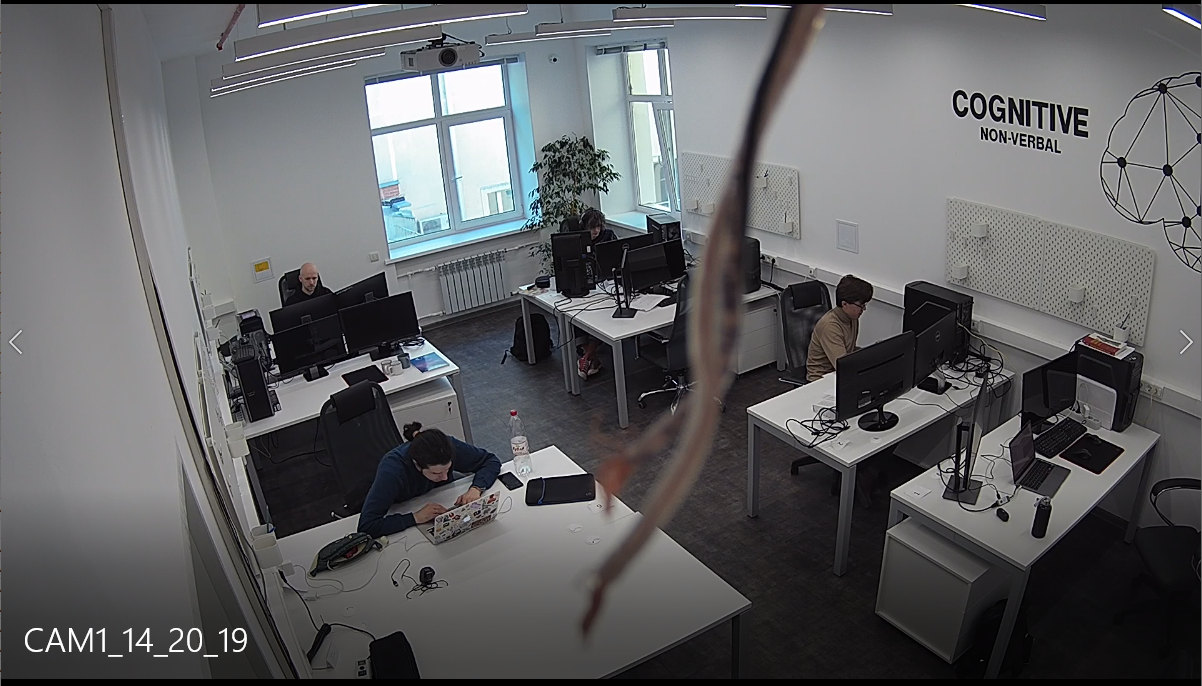
Each subject was informed about his/her participation in the experiment and gave written consent to the processing of personal data. The gender breakdown of the subjects in the experiments was in relation to 5 males to 1 female (pic.14).

Picture 14 – Gender distribution of person tested

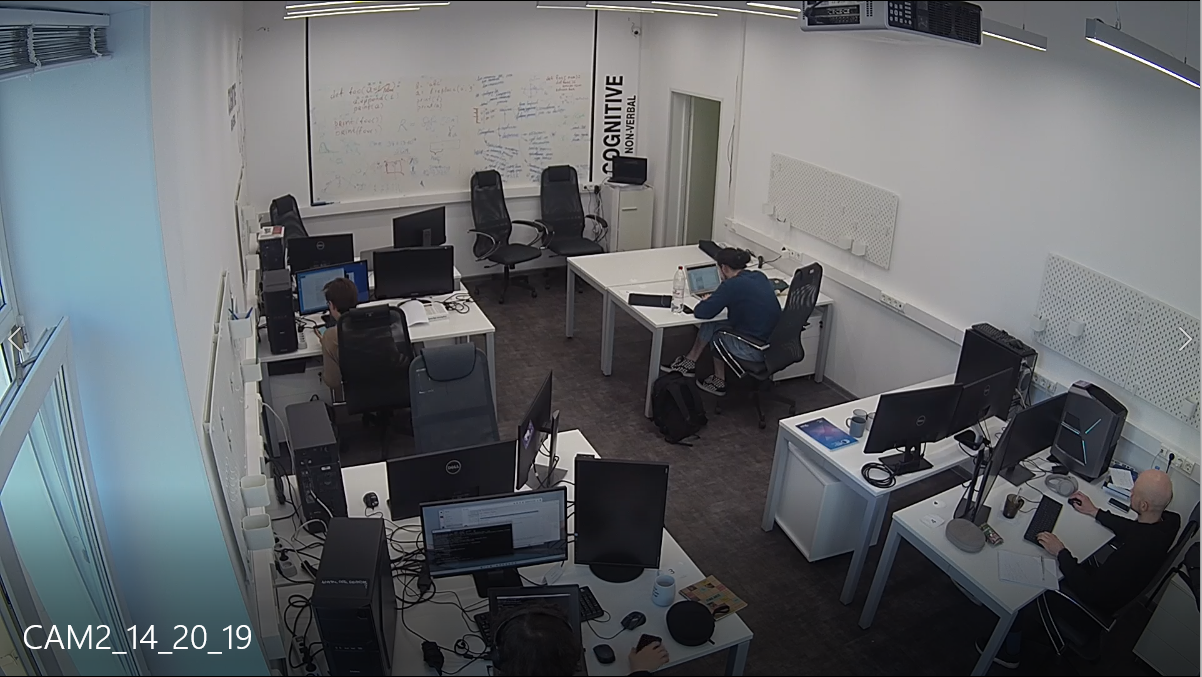
The average age of the subjects was 24.

## 2.2 Data collecting

A total of 17 experimental days were recorded: 16.03.2022 - 22.03.2022, 29.03.2022 and 01.04.2022 - 10.04.2022. The data collected included recordings from near and far IP surveillance cameras (pic.15 and pic.16) as well as data from digital sensors: temperature (C), gas resistance (ohm), relative humidity (%), atmospheric pressure (hPa), altitude (m), luminance window (lux), luminance door (lux).

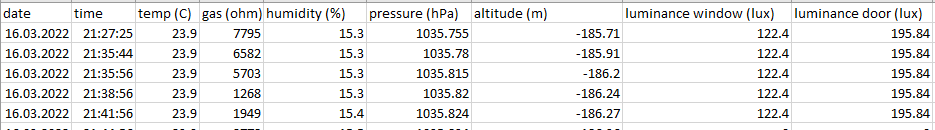


Picture 15 - A shot from a close-up camera 01.04.2022 14:20



Picture 16 - Far camera shot 01.04.2022 14:20

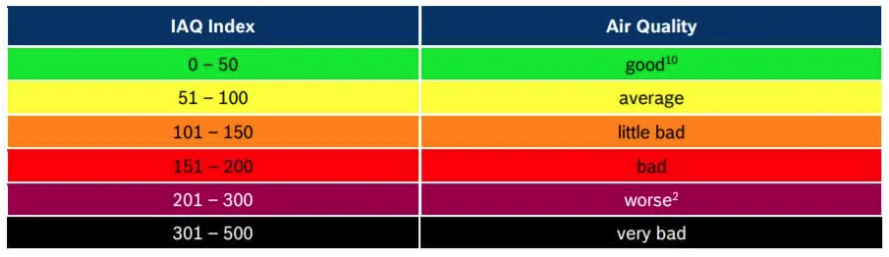
An example of a data frame is shown in pic.17.



Picture 17 - Example of a frame of data received from the sensors

A total of 129.5 hours of .mp4 video and 55634 recordings of sensor readings were collected.

This paper intends to evaluate the psycho-emotional state of a person in relation to environmental factors. It is therefore necessary to convert the sensor values of the Bosch BME680 sensor to a uniform Indoor Air Quality (IAQ) indicator. According to Bosch Datasheet [19] (pic.18) the , the , the .



Picture 18 – relation of IAQ index and Air Quality

The conversion of the values received from the sensor into IAQ [20] units is based on the law of the ideal gas equation:

|  |  |
| --- | --- |
|  | (10) |

where, – gas pressure, Pa;

– volume, ;

- the amount of substance of the gas, moles;

- the absolute temperature, K;

- the gas constant, in SI units .

And the August-Roche-Magnus equations:

|  |  |
| --- | --- |
|  | (11) |

where, – gas pressure, kPa;

- the absolute temperature, K.

Then evaluate the maximum absolute water density:

|  |  |
| --- | --- |
|  | (12) |

where, – is the temperature, C.

Then calculating the absolute humidity:

|  |  |
| --- | --- |
|  | (13) |

where, hum – is relative humidity.

Using relative humidity influence of the present water vapor concentration on the gas resistance can be compensated. The dependency appears to be exponential. Thus, the bare VOC resistance is obtained by:

|  |  |
| --- | --- |
|  | (14) |

where, – is the gas resistance, ohm.

The air quality is calculated as the ratio between (14) and the ceiling value gas\_ceil, which is further squared for a steeper slope at higher air qualities and capped:

|  |  |
| --- | --- |
|  | (15) |

where, – is 0.95 percentile for clear gas, ohm.

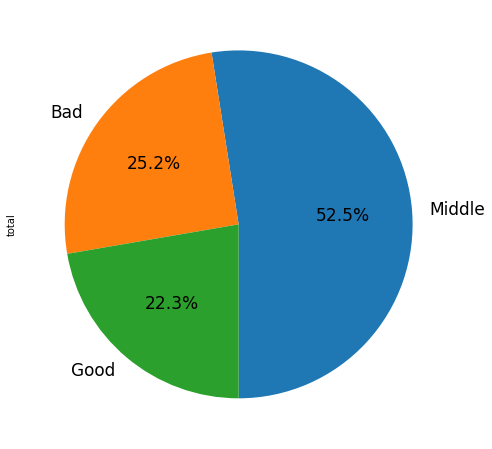
So can be measured:

|  |  |
| --- | --- |
|  | (16) |

After that we can implement IAQ metrics on our dataset with the borders below for the Good ∈[0,100], the Middle ∈(100,200] , the Bad ∈(200,500] we have an air quality distribution (tab. 1) and represent it in pie diagram (pic.19)

Table 1 – Numeric IAQ distribution condition during the data collection

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 12398 | 29201 | 14032 |



Picture 19 – Pie diagram of IAQ condition during the data collection

We can see that more than 50% of experimental data in collect on middle IAQ condition.

## Data labeling

To train a neural network you need to partition the data. Neural networks can only work with images. Working with video is essentially fractioning the video recording into individual images and running these images through layers of neural network.

I used an open-source tool from DVDVideoSoft to break down the video frame by frame. The direct partitioning itself was done with Roboflow Annotate. Because this service allows you to mark images with BBox and prepare them immediately into a suitable format for YOLO.

In our research, a binary distinction will be made between positive and negative human emotional states. Each of the states is characterized by its own set of non-verbal signs, such as gestures and postures. In different psycho-emotional states, a person highlights different non-verbal signs when interacting with objects or other people.

The positive non-verbal signs and their corresponding gestures and postures are presented in the table 2:

Table 2 - Positive human emotional states and their corresponding gestures.

|  |  |
| --- | --- |
| The emotional state of the person | Gestures, poses |
| Sincerity | Open hands, palms up. |
| Consideration of the decision | Person sitting on the edge of a chair, leaning forward, head slightly tilted and resting on the hand |
| Benevolence | A soft smile, slight inclination of the head towards the other person, an expression in the eyes. |
| Interest | A squinting gaze is accompanied by a slightly raised eyebrow or a smile.  Leaning towards the other person (courtesy, attention) |

Negative non-verbal signs and their corresponding gestures and postures are presented in Table 3:

Table 3 - Negative emotional states of a person and their corresponding gestures.

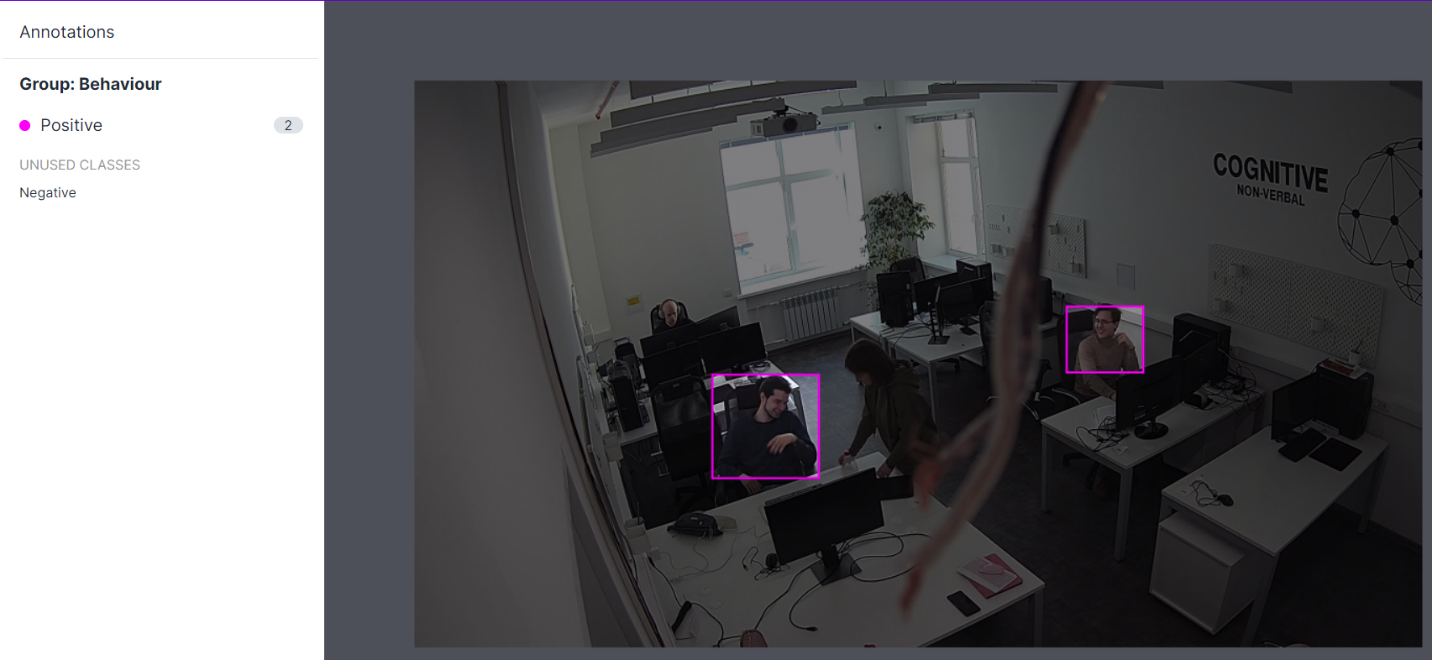
|  |  |
| --- | --- |
| The emotional state of the person | Gestures, poses |
| Feelings of self-blame, strained perceptions of the situation | Hands hidden (behind back, in pockets) |
| Defence or protection | Arms crossed on chest |
| Fatigue or emotional or physical tension | Yawn |
| Desire to distance oneself from one's surroundings | Cross-legged pose |

As an example of positive class markup, consider pic. 20.

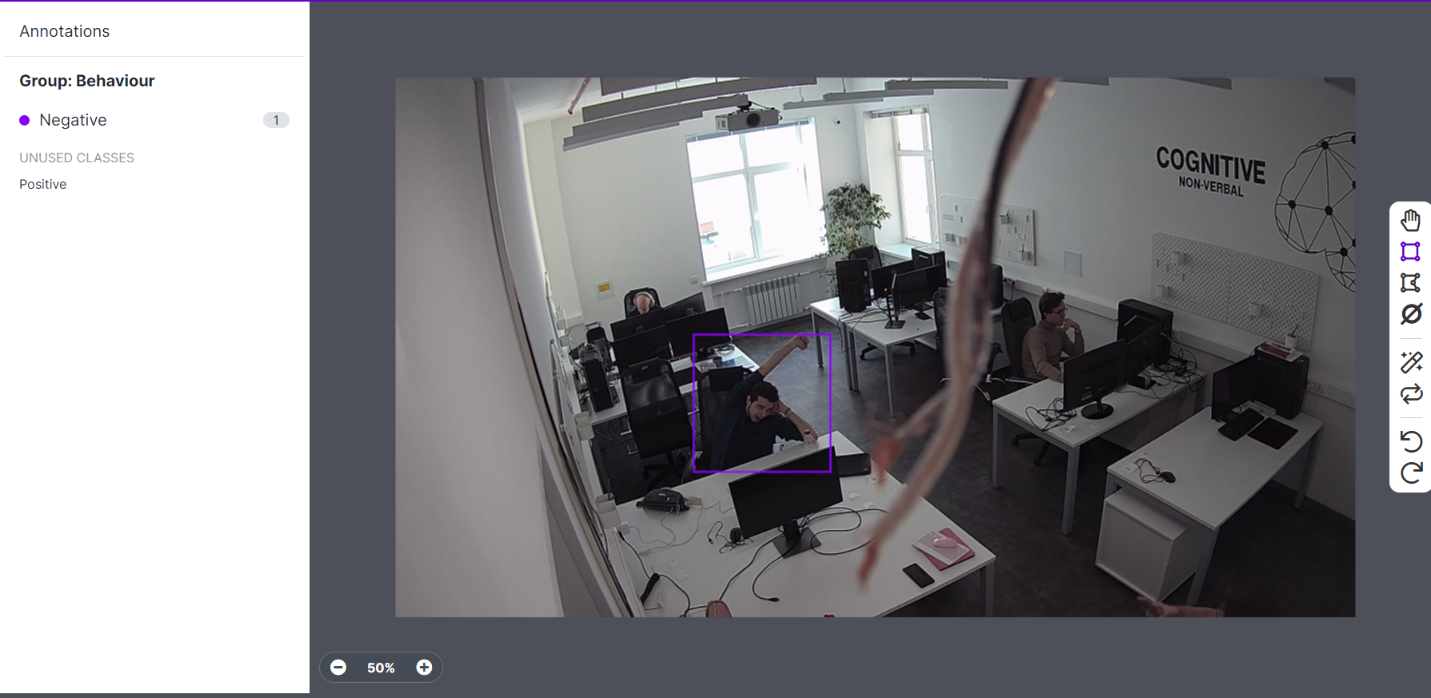


Picture 20 – Positive emotions of people. Man 1 and 2 are smiling.

Thus we can classify people's emotions at manual marking as positive pic. 21.

 Picture 21 – Hand-labeling smiling people to positive class

As an example of negative class markup, consider pic. 22



Picture 22 - Hand-labeling yawn people to negative class

Empirically, it has been found [21] that YOLO training requires an average of 500 photos per pattern. Taking this into account, 4000 photos need to be manually partitioned for this job.

This task is currently at the solution stage.

# Conclusion

During this paper the theoretical foundations of neural networks have been explained. The basic architecture of convolutional neural networks Convolutional->Pooling->Full-connection. We have examined in detail the convolutional operation, as well as the influence of core size on convolutional and pooling operations. Described the purpose of the Full-connection layer at the output of the neural network. We studied basic metrics used in computer vision tasks and theoretically derived the IoU metric. After that, we got acquainted with the structures of models used for object detection tasks. We gave examples of one-stage and two-stage detectors. Using Microsoft COCO dataset as an example we have considered the best practices of neural network models and identified two basic architectures for our study: YOLOR and Scaled-YOLOv4. After further investigation of these architectures we chose Scaled-YOLOv4. We discussed the Scaled-YOLOv4 architecture in detail and also introduced the concept of Transfer-Learning and described the competitive advantages of this approach. We have analyzed in details the algorithm for tracking objects and their centres, and briefly touched the detection of objects through the Kalman filter.

The practical part of this paper concerns the methodology of the experiment and the location of the experiment. The experiment was conducted in the 408 laboratory of the ITMO National Center for Cognitive Development. Two diametrically located IP cameras were used for this study, as well as an illumination sensor together with a Bosch BME680 environmental control sensor, which allows to measure such parameters as air resistance and humidity, temperature and atmospheric pressure. Six people participated in this experiment as subjects. The average age of the subjects was 24 years, and the gender distribution was 5 to 1 in favor of men.

In total, data collection was conducted over 17 experimental days. During this time we managed to collect 129.5 hours of video from and 55634 records of sensor values. It is important to note that this is the volume of raw data, and after data mining their volume will probably decrease.

Because the Bosch BME680 sensor does not calculate Indoor Air Quality directly, we had to manually translate the sensor readings into the IAQ index. The formula for the conversion was derived based on the law of the ideal gas equation and the Magnus equation. We obtained that 52.5% of the experimental data were made in medium quality air, 25.2% in low quality air, and 22.3% in high quality air.

The neural networks do not have the ability to work directly with video data, so we split the video frame by frame and took every 30th frame in order to reduce the amount of data needed to be partitioned while preserving most of the information. Video framing was performed by an open-source tool from DVDVideoSoft.

We decided to use a binary classification of human emotional states in this study: positive and negative. For the positive state we identified such behavioral markers as: Sincerity, Consideration of the decision, Benevolence, Interest. For the negative state we identified the following markers: Feelings of self-blame, strained perceptions of the situation, Defence or protection, Fatigue or emotional or physical tension, Desire to distance oneself from one's surroundings. For each behavioral condition we identified unique nonverbal signs and began to hand mark up the data. Manual partitioning of the data is done using the conditionally free Roboflow Annotate tool.

Empirically, it has been found that YOLO training requires an average of 500 photos per pattern. Taking this into account, 4000 photos need to be manually partitioned for this job.

At the moment, our research is at the stage of manual markup of images based on nonverbal signs.

# References

1. Li Z. et al. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects // IEEE Transactions on Neural Networks and Learning Systems. 2021. P. 1–21.

1. 5.1 Архитектура сверточной нейронной сети [Electronic resource] // StudFiles. URL: https://studfile.net/preview/6871496/page:8 (accessed: 18.06.2022).

1. Сверточные нейронные сети — Викиконспекты [Electronic resource]. URL: https://neerc.ifmo.ru/wiki/index.php?title=Сверточные\_нейронные\_сети (accessed: 18.06.2022).

1. How Do Convolutional Layers Work in Deep Learning Neural Networks? [Electronic resource] // Machine Learning Mastery. 2019. URL: https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/ (accessed: 19.06.2022).

1. Голиков И. Сверточная нейронная сеть, часть 1: структура, топология, функции активации и обучающее множество // Habr. 2018.

1. URL: https://glassboxmedicine.files.wordpress.com/2019/02/confusion-matrix.png?w=1200 (accessed: 19.06.2022).

1. Jiao L. et al. A Survey of Deep Learning-Based Object Detection // IEEE Access. 2019. Vol. 7. P. 128837–128868.

1. Yao Y., Yang Y., Wang J. Scratch Detection of Aircraft Bell Tube Based on Improved YOLOv4 Framework // 2021 4th International Conference on Signal Processing and Machine Learning. New York, NY, USA: ACM, 2021.

1. Lin T.-Y. et al. Microsoft COCO: Common Objects in Context // Computer Vision – ECCV 2014. Cham: Springer International Publishing, 2014. P. 740–755.

1. Wang C.-Y., Yeh I.-H., Liao H.-Y.M. You Only Learn One Representation: Unified Network for Multiple Tasks [Electronic resource] // arXiv.org. 2021. URL: https://arxiv.org/abs/2105.04206.

1. Wang C.-Y., Bochkovskiy A., Liao H.-Y.M. Scaled-YOLOv4: Scaling Cross Stage Partial Network [Electronic resource] // arXiv.org. 2020. URL: https://arxiv.org/abs/2011.08036.

1. Redmon J. et al. You Only Look Once: Unified, Real-Time Object Detection [Electronic resource] // arXiv.org. 2015. URL: https://arxiv.org/abs/1506.02640.

1. Wang C.-Y. et al. CSPNet: A New Backbone that can Enhance Learning Capability of CNN [Electronic resource] // arXiv.org. 2019. URL: https://arxiv.org/abs/1911.11929.

1. TensorFlow Core [Electronic resource] // TensorFlow. URL: https://www.tensorflow.org/tutorials/images/transfer\_learning (accessed: 19.06.2022).

1. Bochinski E., Eiselein V., Sikora T. High-Speed tracking-by-detection without using image information // 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). IEEE, 2017.

1. Bewley A. et al. Simple online and realtime tracking // 2016 IEEE International Conference on Image Processing (ICIP). IEEE, 2016.

1. Peksa J. Prediction Framework with Kalman Filter Algorithm // Information. 2020. Vol. 11, № 7. P. 358.

1. Алгоритм трекинга объектов в реальном времени с обработкой ошибок // Доклады Белорусского государственного университета информатики и радиоэлектроники. № 6 (76).

1. GmbH B.S. BME680 Datasheet.

1. thstielow. GitHub - thstielow/raspi-bme680-iaq: Basic IAQ calculator for the Bosch bme680 sensor, compensating the humidity dependency and long-term drifts. Outputs a gas quality score on a range of 0-100%. [Electronic resource] // GitHub. URL: https://github.com/thstielow/raspi-bme680-iaq (accessed: 19.06.2022).

1. BeyondCurriculum. Урок №4. YOLOv4. Сбор и обработка датасета. Beyond Robotics // YouTube. 2021.