DOI: https://doi.org/10.15276/hait.06.2023.2
UDC 004.932

Evaluation metrics systematization for 2D human poses analysis models

Svitlana G. Antoshchuk[1)]
ORCID: https://orcid.org/0000-0002-9346-145X; asg@op.edu.ua. Scopus Author ID: 8393582500
Anastasiia A. Breskina[1)]
ORCID: https://orcid.org/0000-0002-3165-6788; anastasia.breskina@gmail.com
[1)] Odessa Polytechnic National University, 1, Shevchenko Ave. Odessa, 65044, Ukraine

ABSTRACT

This paper describes the systematization of evaluation metrics for 2D human pose analysis models. Some of the most popular tasks solved using machine learning (ML) methods are detection, tracking and recognition of human actions for various practical applications. There are a lot of different metrics that allow evaluating the model from one point or another. To evaluate a specific task, a certain set of metrics is used. However, as literature analysis shows, the vast number of metric definitions, as well as the use of different terms and multiple representations of the same ideas, causes problems of interpretation and comparison of different ML models and methods in detecting, tracking, and recognizing human actions. The purpose of this work is to analyze the metrics for evaluating methods for processing 2D human poses in video in order to facilitate the informed choice of the metrics. To improve the objectivity of evaluating the results of empirical studies of existing and newly developed methods and models for detecting, tracking, and recognizing human actions, a systematization of existing metrics into subgroups was proposed, depending on what task they evaluate. Four classes of evaluation metrics were introduced: classification metrics, key point's detection, object tracking, and general metrics. Classification metrics are based on quality evaluation and matching values from predicted bounding boxes with ground truths. Key point's detection metrics are oriented on the quality of found joints of the human body skeleton. Tracking metrics evaluate the object detection on each frame and the correctness of determining its trajectory. General metrics are not specifically related to any of the human 2D pose analysis tasks. The prototype of the application based on suggested metrics systematization, the purpose of which is to help data scientists in formalizing the choice of metrics for evaluating models depending on the ML problem being solved and the application area was developed. To evaluate and demonstrate the metrics, that were suggested in this application, Faster R-CNN, SSD and YOLOv3 object detection models were analyzed and compared in scope of 2D human pose analysis application area. The results of the analysis showed that Faster R-CNN and YOLOv3 have the most accurate responses, although they have the disadvantage of a high False positive rate. The implementation also showed that metrics that based on True negative values are uninformative in scope of working with bounding boxes, because of the specific of application area and inability to calculate True negatives on the image data.

Keywords: Computer vision; neural networks; deep learning; metrics; video processing; 2D; efficiency

***For citation*: Antoshchuk S. G., Breskina A. A. "Evaluation metrics systematization for 2D human pose analysis models". *Herald of Advanced Information Technology*. 2023; Vol.6 No.1: 26–38. DOI: https://doi.org/10.15276/hait.06.2023.2**

INTRODUCTION

As technology advances, Machine Learning (ML) is becoming more and more popular in almost every field of human life. To solve ML problems such as classification, anomaly detection, estimation, etc., in connection with the expansion of the capabilities of computers and portable devices, the field of neural networks (NN), and specifically convolution neural networks (CNNs), continues to develop rapidly.

To test the effectiveness of specific ML models and methods, the artificial intelligence (AI) community has developed many datasets that are used to train and test Neural Networks (NNs). Each of them can be used to test the performance of ML models and methods by calculating various performance metrics that reflect the results of their implementation. For instance, there are thousands of different free-to-use public datasets on Kaggle (Available from: https://www.kaggle.com) and comparisons of implementations various scientific research based on almost 8000 datasets on the Papers with Code (Available from: https://paperswithcode.com/datasets) resources [1].

In data science, as in software engineering, a metric is a standard of measure of a degree to which a process possesses some property [2].

Metrics functions and metrics as measurements (the numbers obtained by the application of metrics) are often used as synonyms.

There are many performance indicators, the choice of which depends on such main factors as:

– which specific characteristic of the method is being studied: reliability, speed (response time) and resource intensity;

– ML task (regression, classification, etc.) and in what application area is this task being solved: music source separation, language modeling, computer code generation and analysis, robotics, computer vision, recommendation systems, and others.

Due to these factors and the lack of a clearly structured centralized database of existing metrics, as well as different knowledge of Statistics among data scientists (As Joel Grus wrote, some data scientists are mote statisticians, while others are indistinguishable from software engineers. [3]). Different scientists very often use different terms and formulas for the same metrics in their scientific papers. This significantly complicates the objectivity of comparing the results of testing the performance of existing and newly developed models and methods of ML.

An additional difficulty in choosing performance metrics is the use of different abbreviations for the same metrics and the fact that many abbreviations are remarkably similar to words from human languages, which makes it even more challenging to track from the start.

Since the object of the research is 2D pose processing models, the subject of this research is the validation metrics for these models.

The 2D pose analysis process is divided into several subtasks; each of them has its own set of evaluation metrics:

– Human detection.

– Keypoints detection.

– Object tracking.

– Action classification, pose estimation and forecasting.

The systematization of evaluation metrics for models, that solve these subtasks, is the main goal of this work.

LITERATURE REVIEW

As it was written earlier, in many scientific papers and articles on the Internet, the same metrics are often used in different meanings, depending on how much the author is immersed in software development or Statistics as a subject. There are some good scientific works that shed light on what is stored behind the abbreviations of metrics and what their essence is. However, most of these metrics are considered only in the context of one small specific subtask and are not the main subject of the research itself [1, 2], [4, 5].

Other scientific works are focused on a specific metric [6, 7], its improvement (like the main goal or in scope of other investigation) [8, 9], [10] or the development of new metrics based on it [11].

Since there has been a significant surge of scientific works on the topic of NN and deep learning over the past couple of years, some research teams have created projects on the topic of designing the ontology of development of this area and building dependency graphs, based on topics, citations, benchmarks and used metrics [10]. But this is more related to meta-research annotations and organizing information in the Artificial Intelligence (AI) domain.

Another important free and open resource for getting up to date on current ML works is Papers with Code. It allows you to view the latest publications on the ML topic, get acquainted with the most relevant methods and most used datasets. This resource also provides evaluation tables for these methods based on various metrics. This resource is particularly important, because it allows you to keep track of the latest trends in ML and metrics, which is used to form a list of commonly used metrics in the context of two-dimensional people pose processing.

There are also various manuals and documentation for existing frameworks with different metrics calculations implementation (for instance, some manuals for the free and open-source software libraries for ML and AI TensorFlow, which has a particular focus on deep neural networks (DNN)).

Despite these resources containing a large database of metrics, they either do not include a more detailed description of them or do not classify them according to metrics' meaning. They also contain little to no information about metrics' naming intersection aka synonyms. As a result, the resources themselves do an excellent job, but deeper research of the scientific works behind them will raise a problem of using different terminology by the authors of scientific works.

THE PURPOSE OF THE ARTICLE

The purpose of this work is to analyze metrics, which are used for evaluating peoples' 2D pose analysis models, systematize them, and based on this systematization develop an application that will help data scientists determine the metrics depending on their needs.

MAIN PART. EVALUATION METRICS SYSTEMATIZATION

Classification metrics. These metrics are used to evaluate object detection quality. All of them are based on the Confusion matrix and are not connect-

ed to specific class of the object (animal, human, etc.) or a specific stage of 2D human pose analysis. These metrics are used in scope of human segmentation and classification task and action classification, pose estimation and forecasting (for evaluation of detected human body parts or pose description).

Intersection over union (IoU, also known as Average overlap score (AOS) and Jaccard Index (J) [2, 3]): it is an evaluation metric used to measure the Accuracy of an object detector on a particular image. Any algorithm that provides predicted bounding boxes as an output can be evaluated by using this approach [4].

Two parameters are used to calculate (1) this metric (Fig. 1):

– the ground-truth bounding boxes (bounding boxes from the testing set that localize objects);

– the predicted bounding boxes.

$$IOU = \frac{AreaofOverlap}{AreaofUnion}. \quad (1)$$

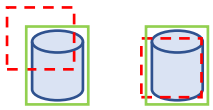


Fig. 1. **Example of IoU: a – IoU =±0.4; b – IoU=±0.9**
Source: **compiled by the authors**

As shown on Fig. 2, Area of Overlap from the IoU formula is the overlap of predicted and ground-truth bounding boxes, and Area of Union is their union.

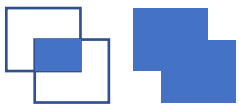


Fig. 2. **Representation of IoU components: a – Area of Overlap: b – Area of Union**
Source: **compiled by the authors**

Success rate: the percentage of frames that has a higher IoU than the threshold [4].

Confusion matrix (also known as an error matrix) [3]: it is used to visualize the performance of a model. It contains number of True positive (TP), False positive (FP, type I error), True negative (TN) and False negative (FN, type II error) predictions.

Precision-Recall curve (PR curve, also known as the receiver operator curve (ROC) curve) (Fig. 3): Metric that summarizes both Recall and Precision. A suitable measure to assess the model's performance on imbalanced datasets [5, 10], [29].

Accuracy: Describes how the model performs across all classes and it is useful when all classes have equal importance [5, 6], [7, 8]. It is calculated (2) as the ratio between True positive and True negative numbers to the total number of predictions:

$$\frac{TP + TN}{TP + TN + FP + FN}. \quad (2)$$

Top-1 Accuracy (Acc@1) means that the model answering with the highest probability must be exactly the True one.

Top-5 Accuracy (Acc@5) means that the model answering with the top five probabilities must contain the expected answer.

3-fold Accuracy is an Average Accuracy over three splits (running model on three different training/testing splits). It is used to measure the final performance.

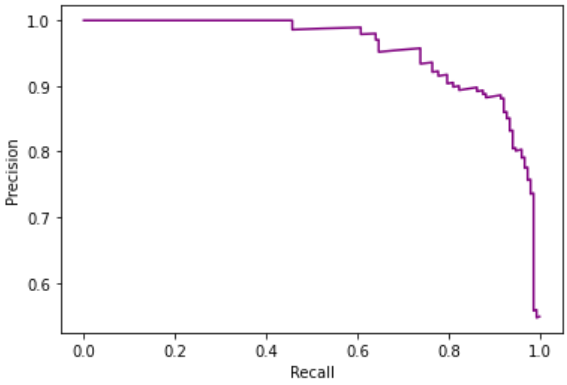


Fig. 3. **An example of the Precision-Recall curve**
Source: **compiled by the authors**

Recall: The Recall score measures the model's ability to detect positive samples [7]. The higher the Recall (3), the more actual positive samples would be detected:

$$\frac{TP}{TP + FN}. \quad (3)$$

Average Recall: is a numerical metric that can be used to compare detector performance [9, 11]. AR (4) is the Recall averaged over all $IoU \in$ [0.5, 1.0]:

$$AR = 2\int_{0.5}^{1} recall(IoU)dIoU. \quad (4)$$

There is a specific notation *max* which means the number of objects for detection:

– AR max=1 means one object for detection per image,

ISSN 2663-0176 (Print)
ISSN 2663-7731 (Online)

– AR max=10 and AR max=100 mean 10 and 100 items, respectively.

Specificity: This metric (5) shows the coverage of actual negative samples [1, 10].

$$\frac{TN}{TN+FP}. \quad (5)$$

Precision: Precision score (6) in scope of working with confusion matrix determines how well a model can find True positives (TP) among all positive predictions (True positives plus False positives (FP)) [11]:

$$\frac{TP}{TP+FP}. \quad (6)$$

In scope of IoU, the Precision P is measured as the distance in pixels between the centers *Cgt* and *Ctr* of the ground truth and the tracker bounding box, respectively.

Average Precision (AP): It is a single number metric that summarizes (7) Precision and Recall [9, 11], [12, 13], [14, 15], [16, 17]. It is done to reduce the impact of the slopes in the curve.

There are several ways to choose the levels of Recall to interpolate the Precision [18].

The traditional one is when 11 levels with 0.1 step are taken:

$$p_{interp}(r) = \max_{\tilde{r} \geq r} p(\tilde{r}),$$
$$AP = \frac{1}{11} \sum_{r \in \{0, 0.1, 0.2, \ldots, 1\}} p_{interp}(r), \quad (7)$$

where p_{interp} is the interpolated Precision and the certain Recall level r is defined as the highest Precision found for any Recall level $\tilde{r} \geq r$.

A new standard chooses all unique Recall levels presented by the data.

$$p_{interp}(r_{n+1}) = \max_{\tilde{r} \geq r_{n+1}} p(\tilde{r}),$$
$$AP = \sum (r_{n+1} - r_n) p_{interp}(r_{n+1}). \quad (8)$$

AP@IoU=.50 or *AP@IoU=.75* notation could be used to specify the IoU value the model will use to determine a positive response. There is a specific way to show the size of detected objects:

– AP small is for small objects with size $< 32^2$ pixels;

– AP medium is for medium objects with size between 32^2 and 96^2 pixels;

– AP large is for large objects with area $> 96^2$ pixels.

Mean Average Precision (mAP). The mAP metric represents the mean of AP values for different classes [18]:

$$mAP = \frac{\sum_{i=1}^{k} AP_i}{K}. \quad (9)$$

Some examples of mAP metric with extra notations:

– *mAPIOU@0.1*: mAP with IoU threshold 0.1 (the model will say that there is an object if IoU=0.1 and higher).

– *mAP@[.5:.95]*: Average mAP over IoU thresholds from 0.5 to 0.95 with step 0.05.

– *AvgmAP(0.3:0.7):* The same Average mAP over different IoU thresholds: from 0.3 to 0.7 with step 0.1.

Normalized Precision: Helps to solve the problem of the error influence on objects with the different size in scope of trackers ranking [10]. Instead of computing the absolute error in pixels, it calculates errors relative to the object size. These errors are then plotted in the range [0, 0.5] and the area under this curve is called Normalized Precision.

Mean Precision: May be confused with mAP and means just the mean value of the Precision score.

F1 Score (also known as Harmonic Precision-Recall Mean, F-measure, F1 measure): It was designed to be a useful metric when classifying between unbalanced classes or other cases with simpler metrics could be misleading [10, 29]. You can also find F1-score (10) Canonical and Augmented metrics.

These ones are connected to variants of supervised learning settings:

– Canonical. The training, validation, and testing sets are from the same dataset.

– Augmented. In this case 20% of a given dataset is for testing and augment the remaining 80% with the other three datasets to form an augmented training and validation dataset.

– Transfer. Three datasets are used for training and validation and selected dataset for testing the learned models

$$2\frac{\mathrm{Pr}ecision \times \mathrm{Re}call}{\mathrm{Pr}ecision + \mathrm{Re}call},$$
$$\frac{2TP}{2TP + FP + FN}. \quad (10)$$

Area under the curve (AUC): It is the average of the success rates corresponding to the sampled overlap thresholds. AUC provides an aggregate measure of performance across all possible classification thresholds. This metric has been criticized;

calling the AUC metric's usage an "equivalent to using different metrics to evaluate different classification rules" [28].

Crowd index: The metric which shows how crowded is an arbitrary image and describes how many people appear in the bounding box of each person [21].

There are notations by Crowd Index of the dataset: Easy (0-0.1), Medium (0.1-0.8) and Hard (0.8-1)

$$CrowdIndex = \frac{1}{n}\sum_{i=1}^{n}\frac{N_i^{\,b}}{N_i^{\,a}}, \qquad (11)$$

where n is a total number of people on the image, $\frac{N_i^{\,b}}{N_i^{\,a}}$ is a crowd ratio, $N_i^{\,a}$ is a number of joints of a i-person in i-person's bounding box and $N_i^{\,b}$ is the number of joints of other persons in i-person's bounding box.

Keypoints detection metrics. These metrics are used to evaluate methods of key point's detection of human bodies (joints of the human skeleton, head, etc.) in scope of key point's detection and 2D pose estimation tasks evaluation. For instance, they are used in human pose detection with COCO-Whole Body dataset (Available from: https://github.com/jin-s13/COCO-WholeBody).

Percentage of correct parts (PCP): It considers a body part to be correctly localized if the predicted end points of the body segments are within 50 % of the ground segment length from their true location [30]. Foreshortened body parts must be localized with greater Precision to be considered correct. PCPm uses 50 % of the average ground truth segment length.

Percentage of detected joints (PDJ): Measures the distance between the predicted and the True joint within a certain fraction of the torso diameter [31].

Percentage of correct keypoints (PCK): It determines the Accuracy of localization of body joints [29, 30]. It measures if the predicted keypoint and the True joint are within a certain distance threshold.

PCKh or PCKh@0.5measures the 50 % of the length of the head segment [30].

PCKh@0.2 is used when a distance between predicted and True joint is less than 0.2 torso diameter.

Object keypoint similarity (OKS): OKS is commonly used in different keypoint challenges and shows how close a predicted keypoint is to the true keypoint [31].

$$OKS = \frac{\sum_i \exp\left(\frac{-d_i^2}{2s^2k_i^2}\right)\delta(v_{i>0})}{\sum_i^{\delta}(v_{i>0})}, \qquad (12)$$

where k_i is the Euclidean distance between the ground truth and predicted keypoint; s is the square root of the object segment area; k_i is the per-keypoint threshold constant (Available from: https://cocodataset.org/#keypoints-eval); v_i is a visibility flag that can be 0 for 'not labeled', 1 for 'labeled, but not visible' and 2 for 'labeled and visible'.

Object tracking metrics. These ones are oriented on tracking, but also include combined metrics, that validate all steps for object tracing process. They are based on Classification metrics.

Some basic terms, that is used in scope of object tracking methods validation metrics [33]:

– False negative or Misses;

– False positive;

– Merge or ID switch (IDSW): when objects get confused after their trajectories (Traj.) intersect;

– Deviation: when an object track is reinitialized with a different ID;

– Fragmentation: when an object stops getting tracked, but his ground truth track still exists.

Trajectory similarity score (Str): It is the sum of the spatial intersection of bounding boxes across the whole trajectories, divided by the sum of the spatial union of BB across the entire trajectories. This one is intricately connected to the IoU metric. A predicted trajectory (prTraj) is True positive (TPTr) and is matched with a ground trajectory (gtTraj) if it has the highest confidence score of all prTraj-s with Str equal or more than a threshold (αtr) [32].

Track-mAP: This one match predictions and ground truth at a trajectory of detected objects [32]. It is calculated in the same way as mAP, except that trajectories are used instead of bounding boxes of detected objects.

Track-mAP requires a trajectory similarity score (Str) between trajectories and a threshold αtr. Trajectories are matched only if the trajectory similarity score is greater than the threshold.

Str is the sum of the spatial intersection of the boxes across the whole trajectories, divided by the sum of the spatial union of the boxes across the entire trajectories. "The most commonly used similarity metric for 2D boxes; 3D boxes and segmentation masks are IoU" [32].

Multiple objects tracking Precision (MOTP): Is the average similarity score (S) over the set of TPs [32]:

ISSN 2663-0176 (Print)
ISSN 2663-7731 (Online)

$$MOTP = \frac{1}{|TP|}\sum\nolimits_{TP}^{S} S \,. \quad (13)$$

Multiple objects tracking Accuracy (MOTA): It measures the overall Accuracy of both the tracking and detection and connected to predict detections value (prDets) and ground truth detection (gtDets) [32]:

$$MOTA = 1 - \frac{|FN| + |FP| + |IDSW|}{|gtDet|}, \quad (14)$$

The Identification metrics (IDF1): It calculates a bijective mapping between the sets of gtTraj-s and prTraj-s [32] of Identities (IDTP, TDTN, etc.):

$$ID - \mathrm{Re}\,call = \frac{|IDTP|}{|IDTP| + |IDFN|}, \quad (15)$$

$$ID - \mathrm{Pr}ecision = \frac{|IDTP|}{|IDTP| + |IDFP|}, \quad (16)$$

$$IDF1 = \frac{|IDTP|}{|IDTP| + 0.5|IDFN| + 0.5|IDFP|} \quad (17)$$

Higher-order tracking Accuracy (HOTA): A combination of three IoU scores for detection (Det-IoU), association (Ass-IoU) and localization (Loc-IoU) [32].

When we talk about Ass-IoU TP, FP, TN, FN are called True positive associations (TPA), False positive Associations (FPA), etc. (see an example of the tracking validation on Fig. 4) [32].

Localization Accuracy (LocA) is an average Loc-IoU over all pairs of matching predicted and ground-truth detections in the dataset [32]:

$$LocA = \frac{1}{|TP|}\sum_{c \in TP} LocIoU(c) \,. \quad (18)$$

Association Accuracy (AssA) is average Ass-IoU over all detections in the whole dataset [32]:

$$Assa = \frac{1}{|TP|}\sum_{c \in TP} AssIoU(c), \quad (19)$$

$$AssA = \frac{1}{|TP|} \times \times \frac{|TPA(c)|}{|TPA(c)| + |FNA(c)| + |FPA(c)|}. \quad (20)$$

The final HOTA formula looks like this:

$$HOTA_{\alpha} = \sqrt{DetA_s \cdot AssA_{\alpha}}, \quad (21)$$

$$HOTA = \int_{0<\alpha\le 1} HOTA_{\alpha} \approx \frac{1}{19}\sum_{\substack{\alpha=0.05 \\ a+=0.05}}^{0.95} HOTA_{\alpha}, \quad (22)$$

here α is a threshold for Loc-IoU calculations and $HOTA_{\alpha}$ is HOTA, calculated for a specific threshold.

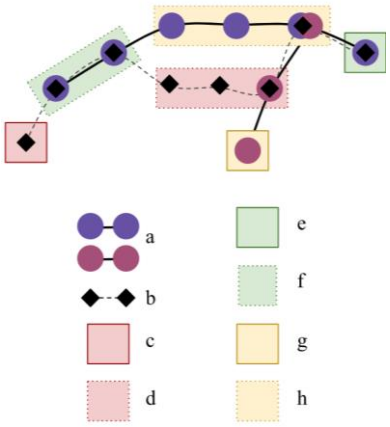


Fig. 4. **Multiple objects tracking basic terms: a – ground-truth detections and trajectories: b – predicted detections and trajectories: c – FP; d – FPA; e – TP; f – TPA; g – FN; h - FNA**

Source: **compiled by the authors**

Here are basic terms that are used in object tracking evaluation process (Fig. 4):

– multi-colored circles, which are connected by an unbroken line (Fig. 4a), are examples of detections and trajectories of objects (GT). At this specific example there are two separate objects with intersecting paths;

– objects detected and tracked by the model are marked with rhombuses connected with dotted line aka predicted trajectory (Fig. 4b), because on this example there is only one detected and tracked object.

Each intersection between prediction and GT or its absence is marked with a separate color and corresponds to such terms:

– there is a predicted object that does not exists in the GT: FP, marked ad rectangle with unbroken borders (Fig. 4c);

– there are three detected object's states (two of them are FP and one is TP) that are falsely connected with trajectory: FPA, marked as red rectangle with dotted border (Fig. 4d);

– the TP detection is marked as green rectangle with unbroken border (Fig. 4e);

– the truly detected object's states connected too right trajectory: TPA, marked as green rectangle with dotted border (Fig. 4f);

– FN object detection is marked as yellow rectangle with unbroken borders (Fig. 4g);

– undetected GT objects and trajectory: FNA, marked as yellow rectangle with dotted border (Fig.4h).

Euclidean error metric (DEucl): Uses predicted trajectory to calculate metric for quantitative evaluation purposes [33]:

$$d_{eucl}(\vec{y},\hat{\vec{y}}) = \sqrt{\sum_{i=1}^{d}(y_j - \hat{y}_j)^2}\,, \qquad (23)$$

where $\vec{y}$ is a GT, $\hat{\vec{y}}$ is a predicted position of the object and d is the number of positions of the tracked object.

Identity switch (IDSW): Counts the number of times that a tracked trajectory changes its matched identity [33].

Tracking success can also be described in terms of these characteristics [34]:

– *Mostly tracked* (MT) if the trajectory is successfully tracked at least in 80 % cases;

– *Mostly lost* (ML) if only the 20 % of trajectory was recovered;

– and *Partially tracked* (PT) in-between.

FM: Counts the number of times that a trajectory in ground is interrupted by the tracking output [34].

General metrics. This type of metrics contains some basic ones, that are not connected to any specific task of 2D pose processing and are related to speed and resource intensity of the model and the specifics of its implementation and training.

Metrics that evaluate the performance of the model include:

– *FPS*: Frame rate [3] (expressed in frames per second) is the frequency of frames which an object detection model can process per second.

– *FLOPs* [3]: How many floating-point operations are required to run a single instance of a given model.

FLOPS: Shows floating-point operations per second and means the computing power of hardware. Unlike the FLOPs (with small‘s’) metric, it helps to evaluate the resource intensity of the developed model [3].

Metrics that show the specifics of the dataset that was used to train and evaluate the model:

– *Extra training data* [4]: informs that more than one dataset was used in the training process.

– *Seen/Unseen* [10]: Is used during different metrics usage. It shows if a metric was calculated basing on the results with seen or unseen dataset categories. Unseen categories are those that do not exist in the training dataset. They are used to evaluate the generalization ability of methods and model.

IMPLEMENTATION

The prototype of a software product Metric Crawler has been developed (Fig. 5) that contains classified information (Fig. 6) about the considered evaluation metrics (Available from: https://metric-crawler.herokuapp.com/). It was created to help data scientists in formalizing the choice of evaluation metrics for their models.

Used technologies: Python, Django, JavaScript and Bootstrap. The application is hosted on Heroku cloud platform service.

Three NN models for object detection (which is the part of human 2D pose analysis process) were evaluated with the suggested metrics.

NN pretrained models were used and tested as an example:

Faster Region-based Convolutional Neural Network (Faster R-CNN, faster_rcnn_resnet50_v1b_coco), Single-Shot Detector (SSD, ssd_512_resnet50_v1_coco) and You Only Look Once version 3 (YOLOv3, yolo3_darknet53_coco). All these models were used only for people detection, other object classes were ignored.

Virtual machine characteristics (standard worker node on Kaggle platform):

73.1 GB HDD, 30GB RAM, 8 vCPU, no GPU acceleration to reduce costs.

Used datasets (both for learning and validation):

COCO 2017 Dataset. Used python runtime environment was in docker, image is gcr.io/kaggle-images/python:v122.

The project was created in as a Python Notebook (Fig. 7, Fig. 8 and Fig. 9). Additional used libraries are pandas, cv2, gluoncv, mxnet, pycocotools.coco, torchvision, seaborn.

Some of metrics have already been implemented in libraries (for instance, IoU, mAP and mAR calculations). Others have been implemented in scope of this work (Fig. 10).

Confusion matrix (Fig. 11) calculation was implemented.

Response of a model assumed to be:

– TP, if IoU ≥ threshold;

– FP, if IoU < threshold, but not zero (the specific of IoU calculation), or when there was any value in response but no GT;

– FN was in that case when bounding box was not detected but supposed to be there (there is GT values in dataset annotations, but no in results);

– TN if there are no data both in annotations and no BB results, which means that there are no people on validation image.

ISSN 2663-0176 (Print)
ISSN 2663-7731 (Online)

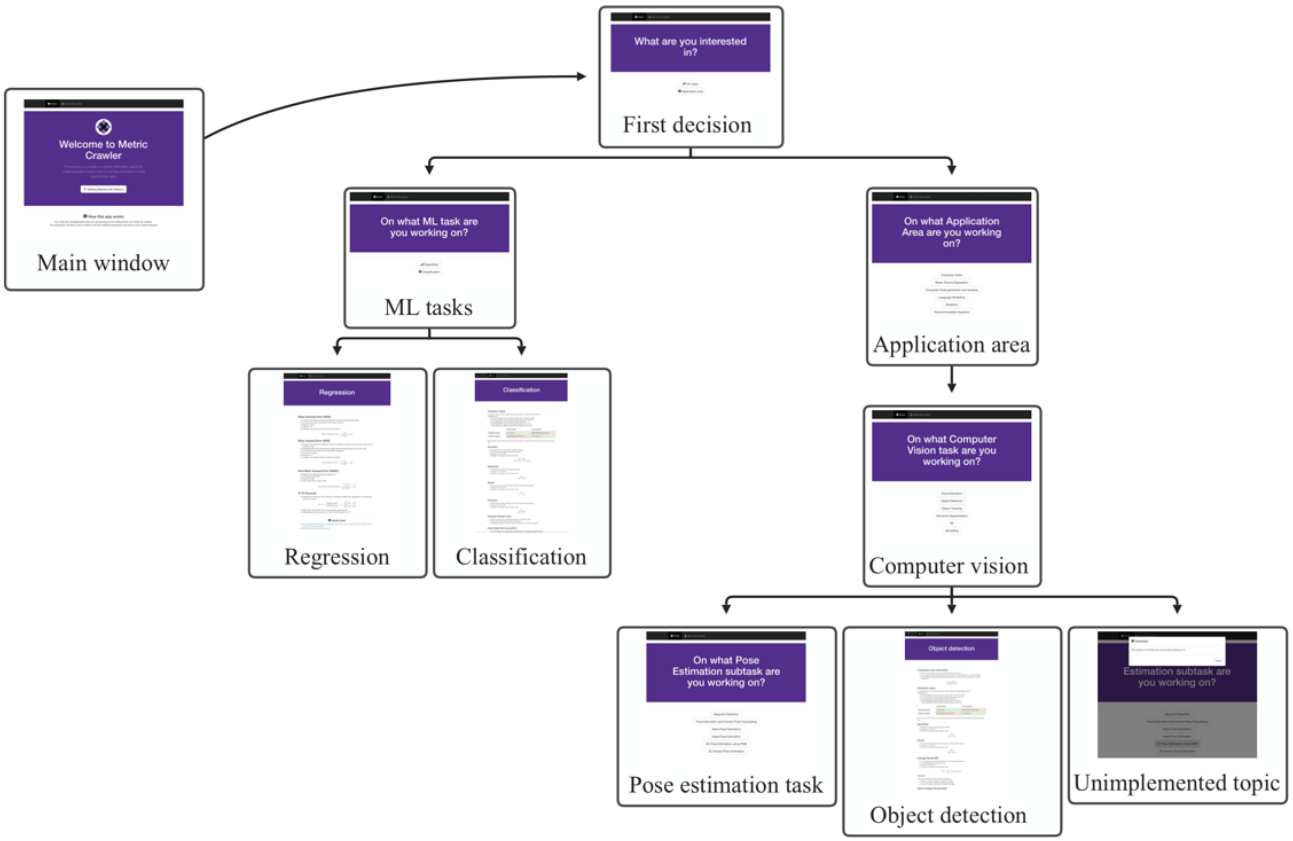


Fig. 5. **The mind map diagram of the Metric Crawler working flow**

Source: **compiled by the authors**

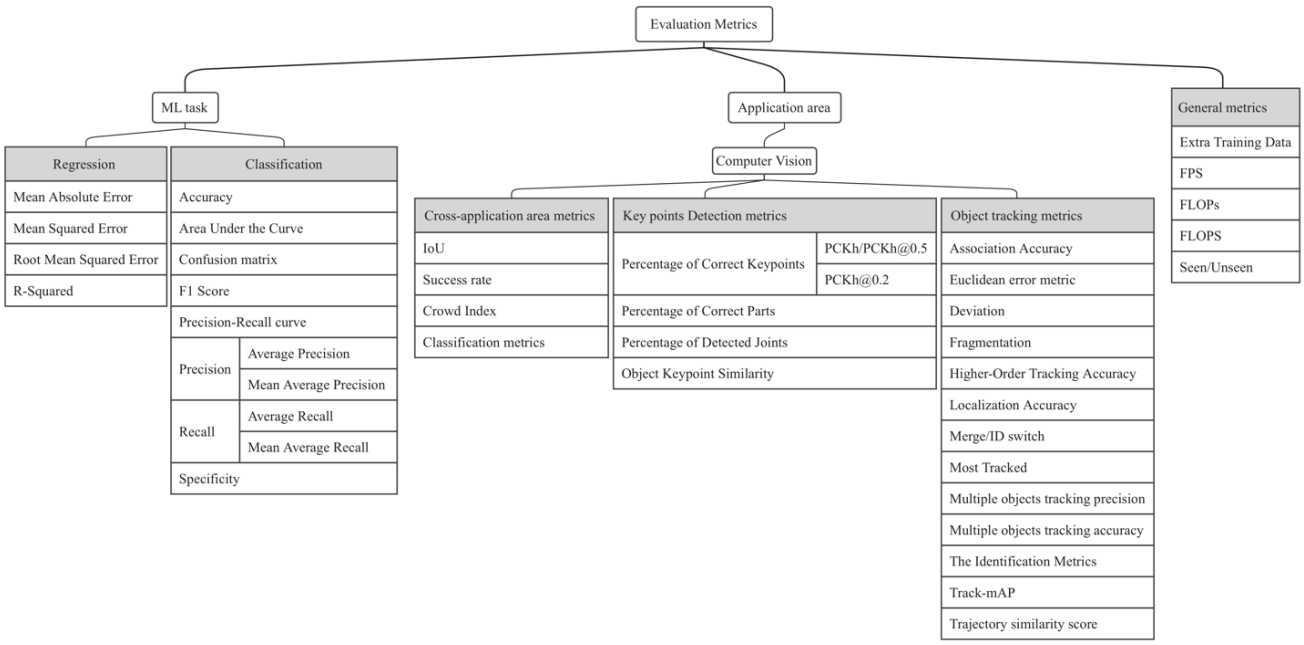


Fig. 6. **Evaluation metrics systematization**

Source: **compiled by the authors**

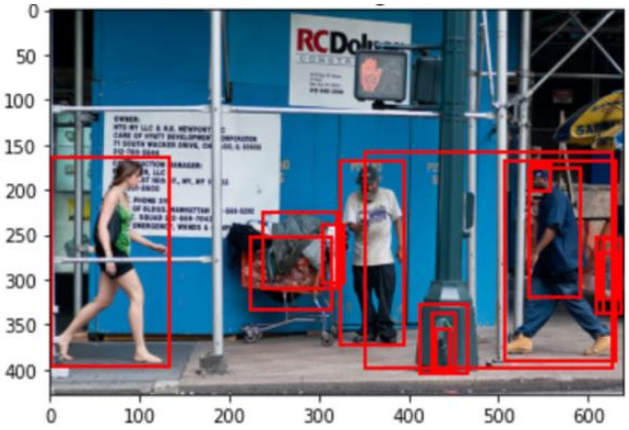


Fig. 7. **Faster R-CNN model result without using IoU threshold**
Source: **compiled by the authors**

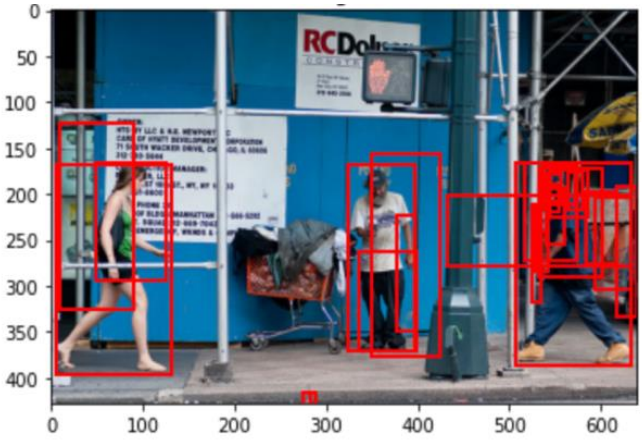


Fig. 8. **SSD model result without using IoU threshold**
Source: **compiled by the authors**

Fig. 9. **YOLOv3 model result without using IoU threshold**
Source: **compiled by the authors**

During the implementation process, some features of calculating metrics in the context of processing results with bounding boxes were clarified.

For instance, in the context of calculating the confusion matrix and further metrics, it is impossible to rely on TN values, since we are working with an image and TN can be located everywhere except TP. This fact makes the TN value uninformative in the context of validation with detection metrics that work specifically with image and bounding boxes (without TP specification in annotations data).

As it was shown on Confusion matrix and Accuracy graphs, all models have low level of FN. Instead of that the problem caused a high amount of FP. As the result, Recall in a standard implementation (without output data filtration) equals 1.

The Precision, F1 and other calculated detection metrics (table 1, 2) also demonstrate a high amount of False positive responses (which worsen the reliability indicators of the considered models). Metrics show that YOLO is better overall, but SSD has the most accurate responses on large objects detection (mAP large and mAR large values are the highest).

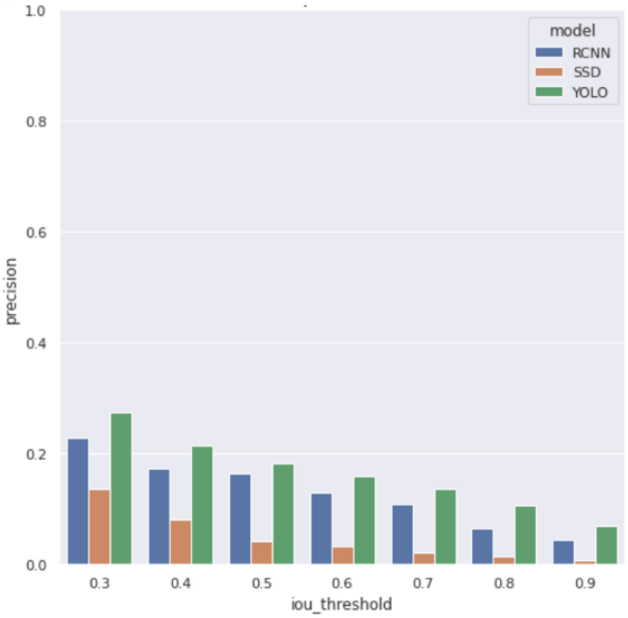
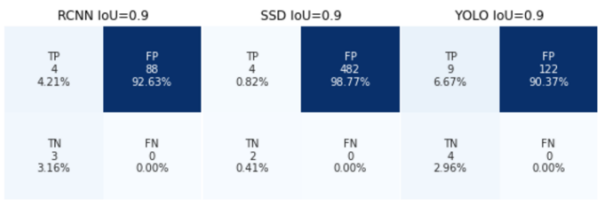


Fig. 10. **Calculated Precision**
Source: **compiled by the authors**



a b c

Fig. 11. **Confusion matrix example for tested NN models with IoU threshold 0.9: a – R-CNN; b – SSD (b); c – YOLO**
Source: **compiled by the authors**

ISSN 2663-0176 (Print)
ISSN 2663-7731 (Online)

Table 1. **Calculated F1 metric based on different IoU thresholds**

Model\Metric	IoU0.3	IoU0.4	IoU0.5	IoU0.6	IoU0.7	IoU0.8	IoU0.9
R-CNN	0.371681	0.296296	0.280374	0.230769	0.196078	0.065217	0.083333
SSD	0.239130	0.148571	0.079051	0.063745	0.040323	0.014403	0.016327
YOLO	0.431138	0.352201	0.309677	0.276316	0.241611	0.106870	0.128571

Source: **compiled by the authors**

Table 2. **Calculated metrics based on the R-CNN, SSD and YOLO models validation**

Metric/Model	R-CNN	SSD	YOLO	Metric/Model	R-CNN	SSD	YOLO
Acc@0.3	0.25263	0.13934	0.29629	mAP medium	0.2826	0.5505	0.7218
Acc@0.9	0.07368	0.01229	0.09629	mAP large	0.7285	0.7710	0.7541
Recall	1	1	1	mAR 1	0.1148	0.1148	0.1333
Precision@0.3	0.22826	0.1358	0.2748	mAR 10	0.2815	0.2889	0.4926
Precision@0.9	0.04347	0.00823	0.0687	mAR 100	0.3074	0.3741	0.5630
mAP	0.2753	0.2905	0.5124	mAR small	0.1375	0.1562	0.4375
mAP50	0.4226	0.4996	0.7939	mAR medium	0.3400	0.5800	0.7200
mAP75	0.2376	0.3065	0.4667	mAR large	0.7333	0.7833	0.7667
mAP small	0.0948	0.0496	0.3196				

Source: **compiled by the authors**

CONCLUSIONS

The article proposes analysis and systematization of the metrics that are commonly used for evaluating 2D human pose analysis models. To facilitate further the empirical studies of existing methods and models for detection, tracking and action recognition, a systematization of existing metrics into subgroups, depending on what task of 2D human pose analysis process they evaluate, was proposed.

Four classes of evaluation metrics were introduced: classification, key point's detection, object tracking metrics, and general metrics. Classification metrics are based on a quality evaluation and matching values from predicted bounding boxes with ground truths. Key point detection metrics are aimed at improving the quality of found joints of the human body. Tracking metrics simultaneously evaluate the object detection on each frame and the correctness of determining its trajectory. General metrics are not specifically related to any of the ML tasks and application area and evaluate the speed, resource intensity and type of training and test datasets. For action classification, pose estimation and forecasting step of human 2D pose analysis classification metrics are being used.

The prototype of the application that used suggested metrics systematization was developed. Its purpose is to help data scientists in formalizing the choice of metrics for evaluating models depending on the ML problem being they solve and the application area they are working in. To evaluate the implemented application and demonstrate the metrics that were suggested Faster R-CNN, SSD and YOLOv3 object detection models were analyzed and compared in scope of 2D human pose analysis application area. Metrics were calculated and compared. This analysis showed that SSD and YOLOv3 have the most accurate responses on large objects detection. However, both have several shortcomings such as high False positive rates on small and medium objects. This implementation also showed inability to calculate True negative values in the context of working with bounding boxes during the metrics calculations, that makes the Recall metric uninformative in certain contexts. This information has been added in the corresponding section of the developed software.

REFERENCES

1. Danelljan, M., Bhat, G., Shahbaz Khan, F. & Felsberg, M. “ATOM: Accurate tracking by overlap maximization”. 2019. DOI: https://doi.org/10.48550/arXiv.1811.07628.

2. Norman E. Fenton & James B. “Software metrics: a rigorous and practical approach”. 2014; 3rd edition. ISBN: 978-1-4398-3822-8.

3. Grus, J. “Data science from scratch: First principles with python”. 2nd edition. 2019. ISBN: 978-1-4919-0142-7.

4. Wu, Y., Lim, J. & Yang, M.-H. “Object tracking benchmark”. *Transactions on pattern analysis and machine intelligence*. 2015; 37 (9): 1834–1848. DOI: https://doi.org/10.1109/TPAMI.2014.2388226.

5. Müller, M., Bibi, A., Giancola, S., Al-Subaihi, S. & Ghanem, B. “TrackingNet: A large-scale dataset and benchmark for object tracking in the wild”. *King Abdullah University of Science and Technology*. 2018. DOI: https://doi.org/10.48550/arXiv.1803.10794.

6. Dhar, S. & Shamir, L. “Evaluation of the benchmark datasets for testing the efficacy of deep convolutional neural networks”. *Visual Informatics.* 2021; 5: 92–101. DOI: https://doi.org/10.1016/j.visinf.2021.10.001.

7. Kishida, K. “Property of Average Precision and its generalization: An examination of evaluation indicator for information retrieval experiments”. 2005. DOI: https://doi.org/10.20736/0000000409.

8. Čehovin Zajc, L., Lukežič, A., Leonardis, A. & Kristan, M. “Beyond standard benchmarks: Parameterizing performance evaluation in visual object tracking”. *International Conference on Computer Vision*. 2017. p. 3323–3331. DOI: https://doi.org/10.1109/ICCV.2017.360.

9. Ren, H., Wang, W., Zhang, K., Wei, D., Gao, Y. & Sun, Y. “Fast and lightweight human pose estimation”. in *IEEE Access*. 2021; 9: 49576-49589. DOI: http://dx.doi.org/10.1109/ACCESS.2021.3069102.

10. Blagec, K., Barbosa-Silva, A., Ott, S. & Samwald, M. “A curated, ontology-based, large-scale knowledge graph of artificial intelligence tasks and benchmarks”. *Sci Data.* 2022; 9: 322. https://doi.org/10.1038/s41597-022-01435-x.

11. Wu, Y., Lim, J. & Yang, M.H., “Object tracking benchmark”. *IEEE Transactions on pattern analysis and machine intelligence.* 2015; 37 (9): 1834–1848. DOI: https://doi.org/10.1109/TPAMI.2014.2388226.

12. Kristan, M. et al. “The sixth visual object tracking vot2018 challenge results”. *ECCV 2018 Workshops. ECCV 2018. Lecture Notes in Computer Science*. 2019; 11129: 3–53. DOI: https://doi.org/10.1007/978-3-030-11009-3_1.

13. Carreira, J., Noland, E., Banki-Horvath, A., Hillier, C. & Zisserman, A. “A short note about Kinetics-600”. 2018. DOI: https://doi.org/10.48550/arXiv.1808.01340.

14. Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzyńska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., Hoppe, F., Thurau, C., Bax, I. & Memisevic R. “The “something something” video database for learning and evaluating visual common sense”. 2017. DOI: https://doi.org/10.48550/arXiv.1706.04261.

15. Soomro, K., Roshan Zamir, A. & Shah, M. “UCF101: A dataset of 101 human actions classes from videos in the wild”. 2012. DOI: https://doi.org/10.48550/arXiv.1212.0402.

16. Lianghua Huang, Xin Zhao & Kaiqi Huang. “GOT-10k: A large high-diversity benchmark for generic object tracking in the wild”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2021; 43 ()5: 1562–1577. DOI: https://doi.org/10.1109/TPAMI.2019.2957464.

17. Fan, H., Lin, L., Yang, F., Chu, P., Deng, G., Yu, S., Bai, H., Xu, Y., Liao, C. & Ling, H. “LaSOT: A high-quality benchmark for large-scale single object tracking”. 2018. DOI: https://doi.org/10.48550/arXiv.1809.07845.

18. Manal El Aidouni. “Evaluating object detection models: Guide to performance metrics”. 2019. - Available from: https://manalelaidouni.github.io/Evaluating-Object-Detection-Models-Guide-to-Performance-Metrics.html. - [Accessed: Jan, 2021].

19. Lin, T., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Lawrence Zitnick, C. & Dollár, P. “Microsoft COCO: Common objects in context”. 2014. DOI: https://doi.org/10.48550/arXiv.1405.0312.

20. Heilbron, F., Escorcia, V., Ghanem, B. & Niebles, J. “ActivityNet: A large-scale video benchmark for human activity understanding”. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015. p. 961–970. DOI: https://doi.org/10.1109/CVPR.2015.7298698.

21. Li, J., Wang, C., Zhu, H., Mao, Y., Fang, H.-S. & Lu, C. “CrowdPose: Efficient crowded scenes pose estimation and a new benchmark”. 2019. DOI: https://doi.org/10.48550/arXiv.1812.00324.

22. Zhang, S.-H., Li, R., Dong, X., Rosin, P. L., Cai, Z., Xi, H., Yang, D., Huang, H.-Z. & Hu, S.-M. “Pose2Seg: Detection free human instance segmentation”. 2019. DOI: https://doi.org/10.48550/arXiv.1803.10683.

23. Zeng, N. “An introduction to evaluation metrics for object detection”. 2018. - Avaliable from: https://blog.zenggyu.com/en/post/2018-12-16/an-introduction-to-evaluation-metrics-for-object-detection/. – [Accessed: Nov, 2022].

24. Danelljan, M., Bhat, G., Shahbaz Khan, F. & Felsberg, M. “Accurate tracking by overlap maximization”. 2019. DOI: https://doi.org/10.48550/arXiv.1811.07628.

25. Simone Teufel. “Information retrieval, computer science tripos part II. lecture 5”. – Avaliable from: https://www.cl.cam.ac.uk/teaching/1415/InfoRtrv/lecture5.pdf - [Accessed: Nov, 2022].

26. Gygli, M., Grabner, H., Riemenschneider, H. & Van Gool, L. “Creating summaries from user videos”. *Computer Vision – ECCV 2014*. 2014; 8695: 505–520. DOI: https://doi.org/10.1007/978-3-319-10584-0_33.

27. Yale Song, Vallmitjana, J., Stent, A. & Jaimes, A.“TVSum: Summarizing web videos using titles”. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2015. p. 5179-5187. DOI: https://doi.org/10.1109/CVPR.2015.7299154.

28. “Google Foundational courses: Classification: ROC Curve and AUC”. – Avaliable from: https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc. – [Accessed: Nov, 2022].

29. Hand, D. J. “Measuring classifier performance: A coherent alternative to the area under the ROC curve”. *Machine Learning.* 2009; 77: 103–123. DOI: https://doi.org/10.1007/s10994-009-5119-5.

30. Andriluka, M., Pishchulin, L., Gehler, P. & Schiele, B. “2D human pose estimation: New benchmark and state of the art analysis”. *2014 IEEE Conference on Computer Vision and Pattern Recognition.* 2014. p. 3686–3693. DOI: https://doi.org/10.1109/CVPR.2014.471.

31. Christian, S. & Alexander, T. “DeepPose: Human pose estimation via deep neural networks”. *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014. DOI: https://doi.org/10.48550/arXiv.1312.4659.

32. Luiten, J., Ošep, A. & Dendorfer, P. “HOTA: A higher order metric for evaluating multi-object tracking”. *International Journal of Computer Vision.* 2021; 129: 548–578. DOI: https://doi.org/10.1007/s11263-020-01375-2.

33. Katsiaryna, M., Michele, D. & Themis, P. “Data series similarity using correlation-aware measures”. 2019. DOI: https://doi.org/ 10.18653/v1/N19-1149.

34. Anton, M., Laura, L., Ian, R., Stefan, R. & Konrad, S. “MOT16: A benchmark for multi-object tracking”. 2016. DOI: https://doi.org/10.48550/arXiv.1603.00831.

Conflicts of Interest: the authors declare no conflict of interest

Received 15.01.2023
Received after revision 25.03.2023
Accepted 02.04.2023

DOI: https://doi.org/10.15276/hait.06.2023.2
УДК 004.932

Систематизація показників ефективності методів аналізу 2d-пози людини

Антощук Світлана Григорівна [1)]
ORCID: https://orcid.org/0000-0002-9346-145X; asg@op.edu.ua. Scopus Author ID: 8393582500

Брескіна Анастасія Аршавирівна [1)]
ORCID: https://orcid.org/0000-0002-3165-6788; anastasia.breskina@gmail.com
[1)] Національний університет «Одеська політехніка», пр. Шевченка, 1. Одеса, 65044, Україна

АНОТАЦІЯ

Ця стаття присвячена систематизації метрик оцінки моделей для аналізу 2D поз людини. Одними з найпопулярніших завдань, які вирішуються за допомогою методів машинного навчання (ML), є виявлення, відстеження та розпізнавання дій людини для різних практичних застосувань. Існує багато різних метрик, які дозволяють оцінити моделі з того чи іншого боку. Для оцінки конкретного етапу аналізу пози людини (виявлення людини, виявлення ключових точок скелета людини, відстеження об'єктів, класифікація та оцінка пози) використовується певний набір метрик. Однак, як показує аналіз літератури, велика кількість метрик, а також використання різних термінів для представлення одних і тих самих понять, спричиняє проблеми з інтерпретацією та порівнянням різних моделей ML. Метою цієї роботи є аналіз та систематизація метрик для оцінки методів аналізу 2D поз людини, щоб полегшити подальший обґрунтований вибір метрик. Для підвищення об’єктивності оцінки результатів емпіричних досліджень існуючих і нових методів і моделей виявлення, відстеження та розпізнавання дій людини запропоновано систематизацію існуючих метрик на підгрупи залежно від того, які прикладні завдання вони оцінюють. Було введено чотири класи метрик оцінювання: метрики класифікації, виявлення ключових точок, відстеження об’єктів і загальні метрики. Показники класифікації базуються на оцінці якості та зіставленні значень із передбачених обмежувальних рамок із реальними об'єктами. Метрики виявлення ключових точок орієнтовані на якість знайдених суглобів скелета людського тіла. Метрики відстеження оцінюють виявлення об'єкта на кожному кадрі та правильність визначення його траєкторії. Загальні показники конкретно не пов’язані з жодними завданнями аналізу 2D пози людини та використовуються для оцінки швидкодії, ресурсоємності та опису особливостей навчання моделі та використаного набору даних. Розроблено прототип веб-застосунку на основі запропонованої систематизації метрик, мета якого – допомогти науковцям із обробки даних у формалізації вибору метрик для оцінки моделей залежно від розв’язуваної задачі ML та прикладної області застосування розробленої моделі. Щоб оцінити та продемонструвати метрики, які були запропоновані реалізованим прототипом, були проаналізовані та порівняні моделі виявлення об’єктів Faster R-CNN, SSD і YOLOv3 у контексті їх застосування для аналізу 2D поз людей. Результати аналізу показали, що Faster R-CNN і YOLOv3 мають найточніші відповіді, хоча вони мають недолік — високий рівень помилкових позитивних результатів. Реалізація також показала, що метрики, які базуються на справжніх від’ємних значеннях, неінформативні для застосування у контексті роботи з зображеннями через специфіку області застосування та неможливість обчислення справжніх негативних значень на даних зображення (ці значенні можуть знаходитися де завгодно на зображенні).

Ключові слова: комп'ютерний зір; нейронні мережі; глибоке навчання; метрики; обробка відео; 2D; ефективність

ABOUT THE AUTHORS

Svitlana G. Antoshchuk - Doctor of Engineering Sciences, Professor of Information Systems Department. Odesa Polytechnic National University, 1, Shevchenko Ave. Odessa, 65044, Ukraine
ORCID: https://orcid.org/0000-0002-9346-145X; asg@op.edu.ua. Scopus Author ID: 8393582500
***Research field*:** Automated management systems and progressive information technologies; information technology; theoretical and applied aspects of multimedia data processing

Антощук Світлана Григорівна - доктор технічних наук, професор кафедри Інформаційних систем. Національний університет «Одеська політехніка», пр. Шевченка, 1, Одеса, 65044, Україна

Anastasiia A. Breskina - PhD Student of Information Systems Department. Odessa Polytechnic National University. 1, Shevchenko Ave. Odessa, 65044, Ukraine
ORCID: https://orcid.org/0000-0002-3165-6788; anastasia.breskina@gmail.com
***Research field*:** Information technologies; computer vision; deep Learning

Брескіна Анастасія Аршавирівна - аспірант кафедри Інформаційних систем., Національний університет «Одеська політехніка», пр. Шевченка, 1, Одеса, 65044, Україна

ISSN 2663-0176 (Print)
ISSN 2663-7731 (Online)