https://doi.org/10.31891/csit-2024-4-5
UDC 004.9

Tymur ISAIEV, Tetiana KYSIL
Khmelnytskyi National University

METHOD OF CREATING CUSTOM DATASET TO TRAIN CONVOLUTIONAL NEURAL NETWORK

The task of creating and developing custom datasets for training convolutional neural networks (CNNs) is essential due to the increasing adoption of deep learning across industries. CNNs have become fundamental tools for various applications, including computer vision, natural language processing, medical imaging, and autonomous systems. However, the success of a CNN depends heavily on the quality and relevance of the data it is trained on. The datasets used to train these models must be diverse, representative of the task at hand, and of sufficient quality to capture the underlying patterns that the CNN needs to learn. Thus, building custom datasets that align with the specific objectives of a neural network plays a critical role in enhancing the performance and generalization capability of the trained model.

This paper focuses on developing a method and subsystem for generating high-quality custom datasets tailored to CNNs. The aim is to provide a framework that automates and streamlines the processes involved in data collection, preprocessing, augmentation, annotation, and validation. Moreover, the method integrates tools that allow the dataset to evolve over time, incorporating new data to adapt to changing requirements or environments, making the system flexible and scalable.

The process of creating a dataset begins with the acquisition of raw data. The data can come from various sources such as images from cameras, videos, sensor feeds, open data repositories, or proprietary datasets. A key consideration during data collection is ensuring that the samples cover the full range of conditions or classes the CNN will encounter in production. For example, in an object recognition task, it is essential to collect images from diverse environments, lighting conditions, and angles to train the model effectively. Ensuring variability in the dataset increases the model's ability to generalize, reducing the risk of poor performance on unseen data.

Data augmentation is a critical step in building a robust dataset, particularly when the size of the dataset is limited. Augmentation techniques introduce variability into the dataset by artificially modifying the existing samples, thereby simulating a wider range of conditions. This helps the CNN generalize better and prevents overfitting. In essence, it allows the model to experience different perspectives and distortions of the same data, strengthening its adaptability to real-world scenarios.

Annotation involves labeling the data samples with the correct class or category information. Depending on the task, annotations may include bounding boxes for object detection, segmentation masks for semantic segmentation, or class labels for classification tasks. The importance of well-annotated data cannot be overstated, as CNNs rely on this labeled information to understand the relationships between input data and the desired output predictions.

Keywords:CNN, dataset, neural network, Roboflow, data preprocessing, data augmentation, labeling.

Тимур ІСАЄВ, Тетяна КИСІЛЬ
Хмельницький національний університет

МЕТОД СТВОРЕННЯ СПЕЦІАЛІЗОВАНОГО НАБОРУ ДАНИХ ДЛЯ ТРЕНУВАННЯ ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ

Завдання зі створення та розробки спеціальних наборів даних для навчання згорткових нейронних мереж (CNN) є надзвичайно важливим через зростаюче використання глибинного навчання в різних галузях. CNN стали основними інструментами для багатьох застосувань, включаючи комп'ютерний зір, обробку природної мови, медичну візуалізацію та автономні системи. Однак успіх CNN значною мірою залежить від якості та релевантності даних, на яких вона навчається. Набори даних для навчання цих моделей повинні бути різноманітними, відповідати специфіці завдання та мати достатню якість, щоб захопити приховані патерни, які модель має опанувати. Таким чином, створення спеціальних наборів даних, що відповідають конкретним цілям нейронної мережі, відіграє ключову роль у покращенні ефективності та здатності моделі до узагальнення. Ця робота зосереджена на розробці методу та підсистеми для створення високоякісних спеціалізованих наборів даних для CNN. Метою є надання структури, яка автоматизує та спрощує процеси збору даних, їх попередньої обробки, аугментації, анотації та валідації. Більше того, цей метод включає інструменти, які дозволяють датасету еволюціонувати з часом, інтегруючи нові дані для адаптації до змінних вимог або умов середовища, що робить систему гнучкою та масштабованою.

Процес створення набору даних починається з отримання сирих даних. Дані можуть надходити з різних джерел, таких як зображення з камер, відео, сенсорні потоки, відкриті репозиторії даних або власні корпоративні датасети. Важливим аспектом під час збору є забезпечення того, щоб вибірка охоплювала весь спектр умов або класів, з якими CNN зустрічатиметься під час експлуатації. Наприклад, у завданні розпізнавання об'єктів важливо зібрати зображення з різних середовищ, при різному освітленні та під різними кутами для ефективного навчання моделі. Забезпечення різноманітності у датасеті підвищує здатність моделі до узагальнення та знижує ризик поганих результатів на невідомих даних.

Аугментація даних є критичним кроком у створенні надійного датасету, особливо коли його розмір обмежений. Техніки аугментації вносять різноманітність у вибірку, штучно модифікуючи наявні приклади, імітуючи ширший діапазон умов. Це допомагає CNN краще узагальнювати та запобігає перенавчанню. По суті, це дозволяє моделі випробовувати різні перспективи й спотворення одних і тих самих даних, зміцнюючи її здатність адаптуватися до реальних сценаріїв.

Анотація передбачає присвоєння міток зразкам даних із правильною класовою або категорійною інформацією. Залежно від завдання, анотації можуть включати рамки для виявлення об'єктів, маски сегментації для семантичної сегментації або класові мітки для класифікації. Важливість якісно анотованих даних важко переоцінити, адже CNN покладаються на ці мітки для розуміння взаємозв'язків між вхідними даними та очікуваними прогнозами.

Ключові слова: CNN, набір даних, нейронна мережа, Roboflow, попередня обробка даних, доповнення даних, маркування.

Introduction

In recent years, there has been a rapid increase in the need to create custom datasets for training convolutional neural networks (CNNs) to solve practical tasks in various domains. This growth is largely driven by the availability of frameworks that allow users to design and train neural networks without requiring deep mathematical expertise [1–3]. Therefore, building custom datasets to meet specific objectives has become an essential task in the development of neural network models, enabling these systems to perform accurately across diverse applications.

The goal of this work is to develop a method for creating custom datasets and to design a system that optimizes the training process of convolutional neural networks.

To achieve this goal, the following tasks must be done:

– create a training dataset tailored to the specific neural network task;

– design and test a CNN model using the custom dataset;

– implement the dataset creation pipeline in a scalable and automated system.

Custom datasets are fundamental in training neural networks, as the quality of a model's performance depends heavily on the data it learns from. Each dataset must contain samples that represent the real-world conditions the model will encounter. Depending on the domain, these datasets may include images, videos, or sensor data tailored to tasks such as object recognition, sentiment analysis, or medical diagnosis.

In the field of computer vision, datasets often consist of labeled images that help neural networks learn to detect patterns. Creating such datasets involves not only data collection but also careful preprocessing, augmentation, and annotation to ensure the samples are both varied and representative. This ensures that the trained model generalizes well and performs accurately when exposed to new, unseen data.

One widely-used tool for building datasets is Roboflow, a platform that simplifies the process of dataset creation, management, and deployment. Roboflow allows developers to easily collect, organize, and label data for neural networks. It also offers features for data augmentation, such as adjusting brightness, contrast, or rotation, which helps increase dataset diversity and prevents overfitting during model training.

Key Components of the Dataset Creation Method:

1) data collection: Collecting raw data from relevant sources such as cameras, public repositories, or proprietary databases. This step ensures that the dataset reflects the conditions and scenarios the neural network will encounter;
2) preprocessing and augmentation: Applying transformations to improve data quality and simulate different real-world conditions. With Roboflow's tools, images can be scaled, rotated, flipped, or have noise added, making the dataset more robust and varied;
3) annotation and labeling: Tagging each sample with the appropriate class labels. For tasks like object detection or emotion recognition, this could involve drawing bounding boxes or assigning specific categories to images. Accurate labeling is crucial for the CNN to learn meaningful relationships between inputs and outputs.

A custom dataset must be diverse and well-balanced to avoid bias. For example, if one class of images is overrepresented, the CNN may learn to favor that class, resulting in poor performance on other categories. Roboflow helps address this issue by tracking class distribution and suggesting ways to balance the dataset. This ensures the model generalizes well across all categories [7-8].

Creating custom datasets tailored to specific CNN tasks plays a vital role in ensuring the success of neural network models. By using Roboflow, developers can streamline the data collection, preprocessing, and annotation processes, leading to more efficient and scalable workflows. This method not only ensures higher accuracy in model predictions but also enhances the network's ability to generalize across different environments and tasks. As deep learning technologies continue to evolve, the demand for high-quality, task-specific datasets will remain a critical factor in the development of intelligent systems.

One of the key advantages of Convolutional Neural Networks (CNNs) is their ability to automatically extract relevant features from data. This means that instead of requiring a manual selection of features or a detailed analysis of how the input variables are related, CNNs learn to identify the most important patterns and relationships by themselves during training.

For example, in traditional machine learning approaches, a lot of time and effort might be spent analyzing data to understand which characteristics (or features) are most important. In image classification, this could mean manually deciding if color, shape, or texture should be prioritized. With CNNs, this step is not necessary. The network uses its layers to progressively learn which features matter most for the task at hand.

Because of this capability, CNNs reduce the need for a pre-analysis of data correlations. Instead of manually identifying what might be important, the network "learns" by itself through exposure to the training data. This automatic feature extraction saves time, minimizes human error, and often leads to better model performance since CNNs can uncover subtle patterns that may not be immediately obvious to a person.

Analysis of Existing Solutions

The increasing integration of technology in various fields has led to the emergence of numerous online platforms for compiling and sharing diverse datasets, greatly enhancing accessibility and participation for researchers globally. The rise of artificial intelligence, particularly in the domain of image recognition, has made it feasible to automate processes such as the classification of objects, including ancient artifacts. This advancement has prompted the exploration of machine learning techniques for various applications, including the study of historical items, artworks, and even natural specimens. Currently, automating the classification of ancient artifacts often serves more as an academic pursuit than a practical solution due to several key factors:

1) many existing datasets are limited in size, with most models requiring thousands of samples to achieve reliable accuracy. Datasets with fewer than 1,000 images often lead to overfitting, where the model learns noise rather than the underlying patterns;
2) no single method guarantees complete accuracy in classification. This uncertainty is compounded by the variability in the quality of images, including differences in lighting, angles, and backgrounds, which can adversely affect the model's performance;
3) many models struggle with recognizing items that were not included in their training sets. This limitation highlights the importance of diverse training data that encompasses various types, conditions, and contexts in which artifacts may appear;
4) the need for such systems is not commercially driven; collectors or museums often prefer expert analysis over machine learning solutions that still require human verification. This reliance on expertise underscores the complexities involved in understanding the historical context and significance of each item, which cannot always be captured by algorithms.

Despite these challenges, several innovative systems have been developed for classifying various artifacts. For example, a research group from the University of Tokyo designed a CNN specifically for categorizing ancient pottery shards. Their dataset comprised over 15,000 images of pottery, organized into multiple classes based on historical significance, material, and decorative styles. The team employed the VGG16 architecture, utilizing transfer learning to leverage pre-trained weights. After 150 epochs of training, they achieved an impressive classification accuracy of 94%, demonstrating the effectiveness of their approach and contributing valuable insights into the pottery's historical context [21].

In another instance, researchers from Stanford University created a classification model for identifying different species of ancient coins, combining state-of-the-art machine learning with archaeology. They compiled an extensive dataset of 20,000 high-resolution images of coins from various eras, meticulously categorized by denomination, material, and design features. This dataset formed the backbone of their project, enabling the team to train their model using the EfficientNet architecture, chosen for its optimal balance between computational efficiency and accuracy. EfficientNet, with its scalable design, allowed the researchers to efficiently process and analyze the intricate details of each coin, such as inscriptions and fine engravings. Their model achieved a classification accuracy of up to 90%, showcasing how advanced deep learning techniques can provide valuable tools for traditional fields like archaeology. This breakthrough not only aids historians and numismatists in identifying and cataloging ancient coins but also serves as a template for similar studies in cultural preservation, facilitating the digitization and analysis of other historical artifacts.[22]

Furthermore, a team of data scientists in Canada demonstrated the practicality of convolutional neural networks (CNNs) in modern financial systems. Their project focused on identifying and classifying contemporary currency notes, addressing challenges like fraud detection and authenticity verification. Using a dataset of 10,000 images of various currency denominations, they employed techniques such as data augmentation, normalization, and the generation of synthetic data through computer graphics to enhance the dataset’s diversity and robustness. Data augmentation introduced variations in lighting, angles, and backgrounds, simulating real-world conditions. As a result, their CNN achieved an impressive 97% accuracy on validation sets, highlighting the potential of deep learning to improve reliability and efficiency in applications requiring high precision, such as banking and security.[23]

Similarly, a notable project on the collaborative platform Medium focused on developing a mobile application aimed at helping collectors identify rare and valuable coins. This project adopted a user-centric approach, relying on community contributions to build a dynamic dataset. Users uploaded images of their collections, which were then annotated to create a growing dataset tailored to the recognition of rare editions and minting errors. The lightweight CNN architecture powering the app was designed to operate efficiently on mobile devices, balancing computational constraints with robust performance. With a preliminary classification accuracy of 92%, the application not only provides immediate value to collectors but also encourages active user participation, enriching the dataset over time and ensuring the model remains relevant and effective.

Another innovative approach gaining momentum is the use of generative adversarial networks (GANs) to augment limited datasets. GANs, with their ability to generate realistic synthetic data, are proving to be a game-changer for artifact classification tasks. Researchers have employed GANs to create high-quality images of artifacts that mimic real-world conditions, such as varying levels of wear and lighting. This synthetic data complements existing datasets, allowing models to learn from a broader range of inputs and reducing the risks of overfitting.

Preliminary studies using GAN-augmented datasets have reported significant improvements in classification accuracy, as models trained on these datasets demonstrate enhanced generalization to unseen data. This method has the potential to revolutionize the field, particularly for niche applications where acquiring large volumes of real-world data is challenging.

In summary, the field of artifact classification through deep learning is experiencing rapid advancements, but challenges remain. Current research underscores the importance of tailored approaches, with researchers leveraging standard CNN architectures, efficient preprocessing techniques, and cutting-edge innovations like GANs to achieve high performance. However, the lack of a universal solution highlights the need for continued experimentation and refinement. The integration of diverse, high-quality datasets and the exploration of novel methodologies will play a crucial role in overcoming existing limitations. As these technologies mature, their application promises to not only automate classification tasks but also deepen our understanding of cultural artifacts, creating opportunities to preserve and appreciate the richness of human history.

Creating Dataset using Roboflow to train custom model

Creating a custom dataset using Roboflow for an emotions dataset involves a series of methodical steps, from data collection to model training:

1) define objectives;
2) data collection;
3) organize images;
4) create or log into Roboflow account;
5) set up a new project;
6) click on "New Project," provide a project name, and select the appropriate project type ;
7) upload images;
8) label images;
9) apply data augmentation;
10) export the dataset;
11) download additional annotation files, such as CSVs, for better class training.

Initially, first step is to define the objectives of the project. It is necessary to determine the purpose of the emotions dataset, specifically whether to classify emotions in images, videos, or text. For the purposes of this example, it was decided to focus on images representing various emotions, such as happiness, sadness, anger, and surprise.

Next, step is to gather a collection of images that depict these different emotions. During this project it was decided to use authors' own images by taking screenshots from webcam to gather various images. It is essential to ensure that there is a balanced number of images for each emotion class to avoid bias in the model.

Once the step of collecting images is done, it is essential to organize them into folders, each labeled with the corresponding emotion. For instance, creating folders named “happy,” “sad,” “angry,” “surprised,” and “neutral.” This systematic organization facilitates the subsequent labeling process.

After organizing the images, we need to create a Roboflow account by visiting their website and signing up, if we do not already possess an account. Once logged in, click on the “New Project” button, choosing a project name and setting the type of project to “Image Classification,” as this aligns with our current project on emotions dataset (Fig. 1).

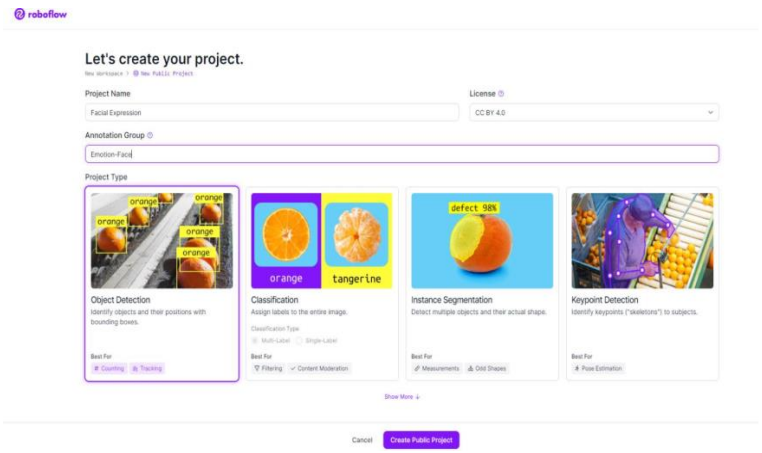

Fig. 1. Settings for creating dataset in Roboflow service

In the project dashboard, find the option to upload images. We can either drag and drop my image folders or use the upload button to select the folders. Roboflow supports bulk uploads, allowing us to upload all the images simultaneously (Fig. 2).

Once the images have been uploaded, proceed to the labeling phase. Roboflow provides tools for labeling images, enabling users to assign labels to each image based on the emotion it represents. This step is critical, as it defines the categories from which the model will learn (Fig. 3).

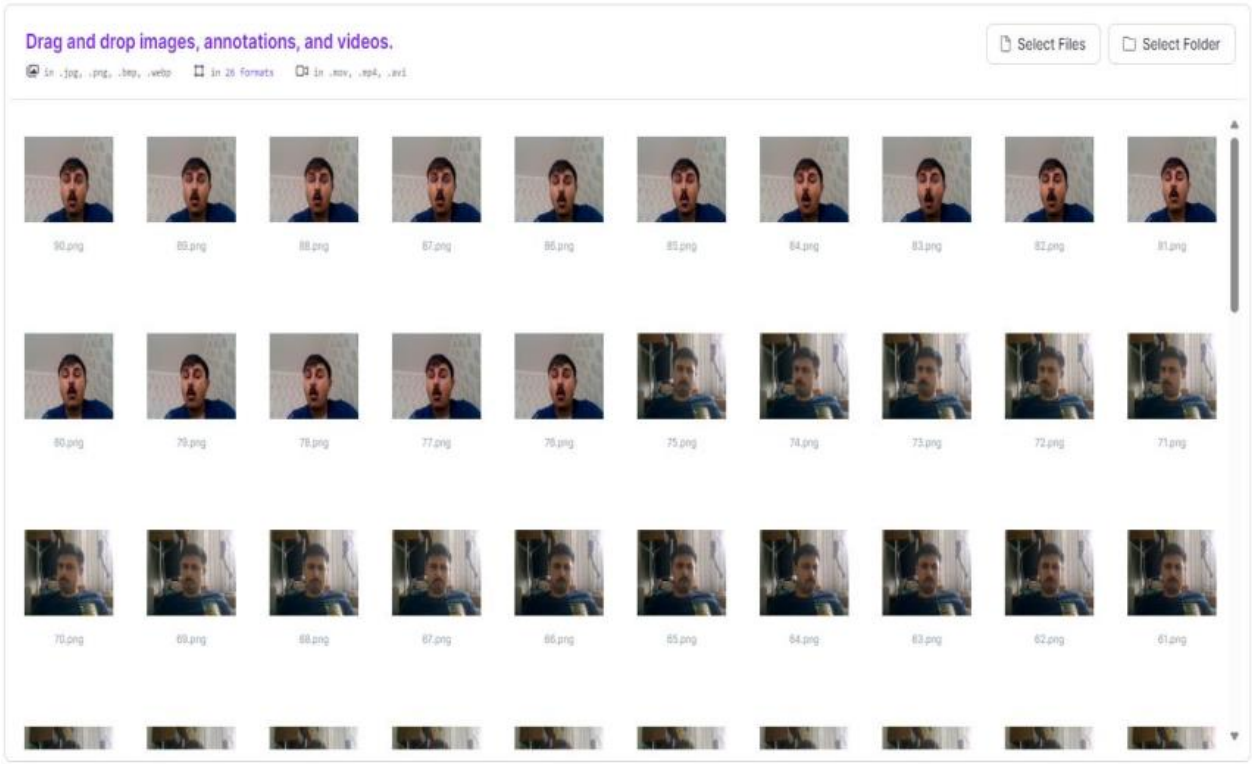

Fig. 2. Process of downloading images to Roboflow

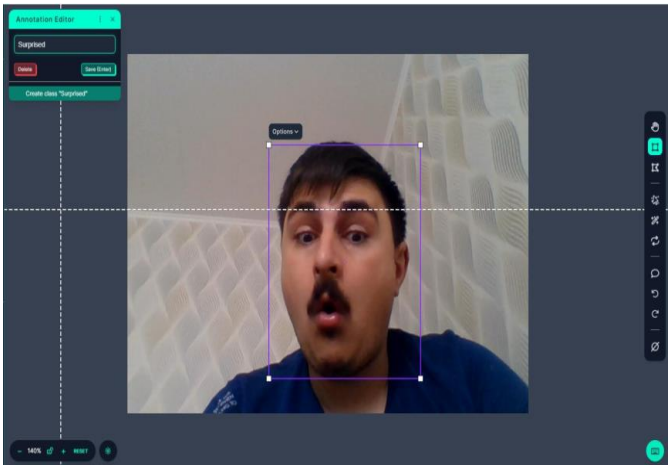

Fig. 3. Process of labeling images

After all images are labeled, advance to the dataset augmentation phase. Roboflow offers various augmentation techniques, such as rotation, flipping, and color adjustments. These techniques enhance the robustness of model, used in this paper, by increasing the diversity of the training data (Fig. 4).

Once we have applied the desired augmentations, we can export the dataset. Roboflow allows users to choose the format for exporting, such as TensorFlow, PyTorch, or YOLO. Users can select the format that is compatible with the framework they intend to use for training custom model model (Fig. 5).

Additionally we can download marking file to help model better determine classes and help it to train more accurately (Fig. 6).

With the dataset exported, the training process begins. Users can import the dataset into their machine learning framework, set up the model architecture, and configure the training parameters. During training, the model learns to classify emotions based on the labeled images that was created using Roboflow and custom method.

Finally, after completing the training, next step is to evaluate the model's performance on a separate validation set to ensure it generalizes well to unseen data. Depending on the results,users may need to fine-tune the model or gather additional data to improve its accuracy. This process of creating a custom dataset using Roboflow

provides a streamlined approach to preparing data for machine learning tasks focused on emotion recognition. Additionally, it is very easy to add new images to the dataset using Roboflow platform.

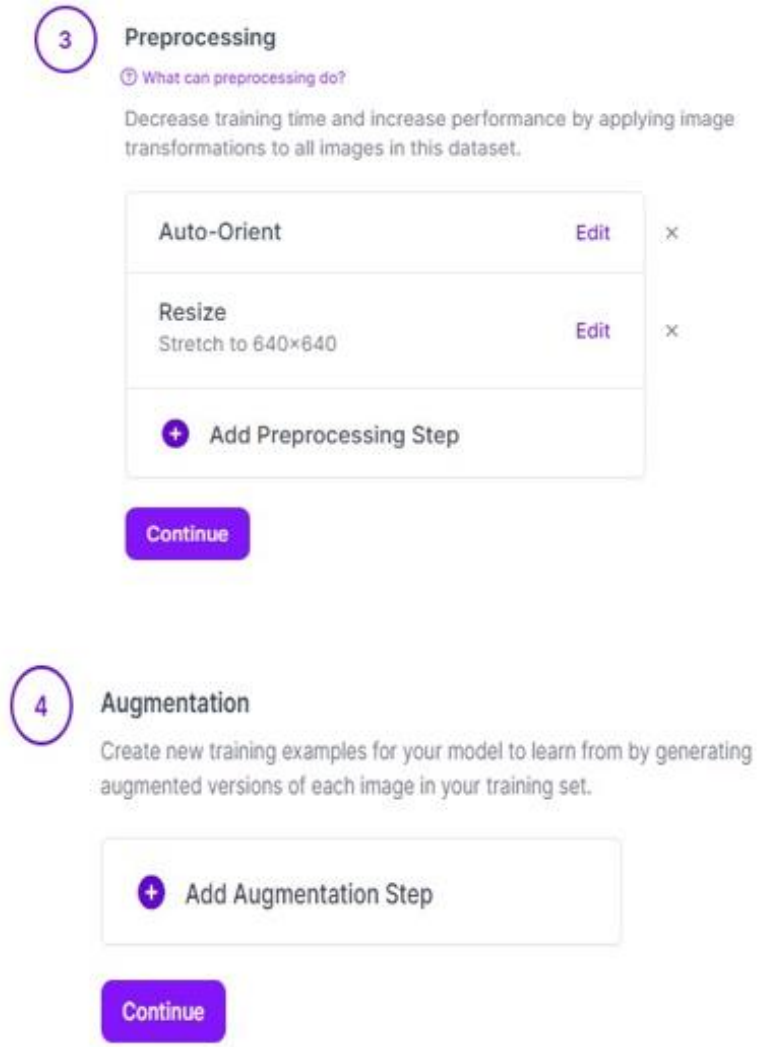

Fig. 4. Augmentation settings of Dataset

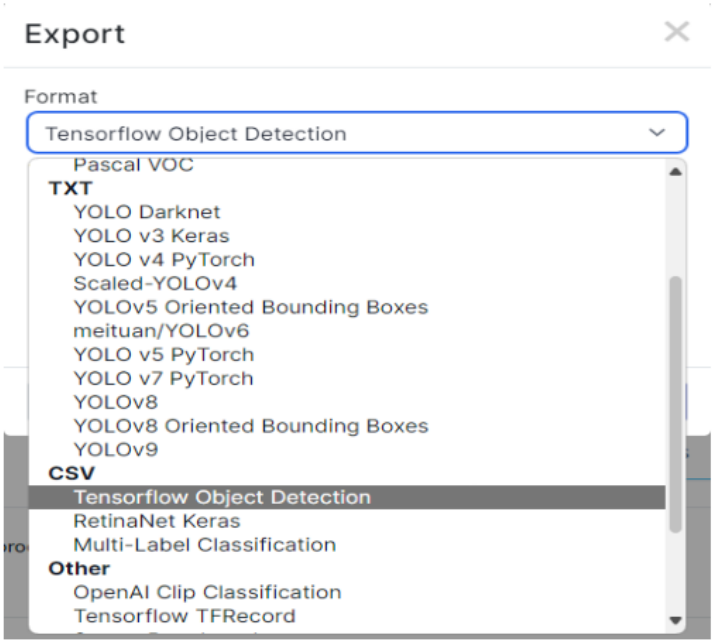

Fig. 5. Process of choosing model to download custom dataset

	A	B	C	D	E	F	G	H
1	filename	width	height	class	xmin	ymin	xmax	ymax
2	24_png.rf.1c3a6423e6ff09cabee828ab199d0973.jpg	640	640	Fear	237	205	464	609
3	71_png.rf.33ef9dc653facc8d7f0eb45361110130.jpg	640	640	Sad	204	105	426	492
4	44_png.rf.1e92bc4aaea0cdf459ad566f292c74bc.jpg	640	640	Happy	226	100	433	492
5	75_png.rf.0dc63957f9857e73a63016ec2a7bf131.jpg	640	640	Sad	232	109	434	481
6	79_png.rf.2a8ae3aef08ecdf99ea437e03a8c398b.jpg	640	640	Surprised	225	133	459	597
7	37_png.rf.2e1a505eec44258800425ff6be616e0a.jpg	640	640	Happy	231	95	428	490
8	captured_image_11_png.rf.18438147f7173ec3bc47cc987c223e7d.jpg	640	640	Angry	230	94	444	529
9	26_png.rf.35e09a23d2cfa65d28abe9f47bb4ffc1.jpg	640	640	Fear	250	177	469	612
10	64_png.rf.257bc1b87707558abd461e0869a785c1.jpg	640	640	Sad	235	96	446	503
11	78_png.rf.0921351535c2e14297d2bc6ecbeb969d.jpg	640	640	Surprised	229	154	457	581
12	2_png.rf.23d802444575b1eab8c79666514ffed0.jpg	640	640	Disgust	195	132	407	509
13	90_png.rf.3b2d8ca2769c342249c78be6a941731a.jpg	640	640	Surprised	224	131	490	617
14	80_png.rf.1929c88bddadb99881ecb353a6b16dd9.jpg	640	640	Surprised	222	125	467	598
15	22_png.rf.23155c8e0f7553bb4314785ad2332e20.jpg	640	640	Fear	248	217	456	606
16	66_png.rf.0536e4b48f804706a1a957e1c7336d75.jpg	640	640	Sad	214	111	431	512
17	73_png.rf.417bb0454af3dd2af2ddcf3611de2cd6.jpg	640	640	Sad	205	99	422	497
18	53_png.rf.2dad3f44c2aa8faffbda051a47eb6463.jpg	640	640	Neutral	235	92	431	490
19	38_png.rf.29a49656ba2d019435d4229ed180f90c.jpg	640	640	Happy	227	97	435	478
20	7_png.rf.3c067b760550d56f861847235f2b7258.jpg	640	640	Disgust	207	119	409	501
21	35_png.rf.23ba17a971af9b45dde49707e73deafc.jpg	640	640	Happy	242	96	422	478
22	87_png.rf.441b69207f71f91840391ec549398b68.jpg	640	640	Surprised	233	133	484	617
23	captured_image_0-checkpoint_png.rf.13a7d3b430bc9c19425684b4577cddf6.jpg	640	640	Angry	205	134	436	505
24	51_png.rf.41ffe9acc171846e13eb8f571c65a81b.jpg	640	640	Neutral	249	123	435	473
25	20_png.rf.492211078e6f6b0f555fa7104573e05c.jpg	640	640	Fear	252	219	452	607
26	65_png.rf.4d2fa50f62afd08326d0e2cf4ba66123.jpg	640	640	Sad	216	102	439	503
27	captured_image_12_png.rf.4fb5e09076693091738367a23acd7df7.jpg	640	640	Angry	226	111	436	530
28	54_png.rf.505575c49c88003db5772995ad230c9a.jpg	640	640	Neutral	241	111	428	466
29	captured_image_8_png.rf.6c75e63a160be7396525537bca6cac20.jpg	640	640	Angry	235	122	447	495
30	27_png.rf.61b91917213d0523289f32fd8845c0d5.jpg	640	640	Fear	239	185	466	590
31	19_png.rf.4bb7e5568845d44c012d2d217f0a4798.jpg	640	640	Fear	237	205	452	580
32	45_png.rf.419ec9f7b764dba6bbf5bc1a35b6fe37.jpg	640	640	Happy	237	102	457	495
33	33_png.rf.5cc6dea371e30696f6d975da2e8e7bf4.jpg	640	640	Happy	232	77	434	487
34	46_png.rf.8736d8150621c8299241beb00e332b9a.jpg	640	640	Neutral	224	103	439	498
35	63_png.rf.76470f830409480b10f6f6745e254afb.jpg	640	640	Sad	223	92	435	489

Fig. 6. Additional CSV file for better training

Conclusions

The custom dataset creation process using Roboflow for emotion recognition provides an end-to-end solution that ensures high-quality data preparation for training machine learning models. It begins with uploading and organizing raw images, followed by accurate labeling of emotions, which lays the foundation for building robust models. Data augmentation techniques applied through Roboflow enhance dataset diversity, improving the model's ability to generalize to real-world scenarios. Once labeled and augmented, the dataset can be exported in multiple formats, compatible with different machine learning frameworks, ensuring flexibility in model development.

This structured approach supports efficient and seamless preparation of data for further training and validation. The system ensures that researchers can focus on optimizing their models rather than dealing with data inconsistencies or formatting issues. Additionally, by enabling integration with frameworks like TensorFlow, PyTorch, or YOLO, Roboflow simplifies the transition from data preparation to model training and deployment.

With the prepared dataset, the trained model can reliably classify emotional states, addressing the challenge of recognizing nuanced human emotions. Such models find applications in areas like sentiment analysis, mental health monitoring, and human-computer interaction. Overall, Roboflow provides a streamlined solution to manage the complexities of dataset creation, contributing to the development of more accurate and reliable emotion recognition models and opening new opportunities for real-world use cases.

It is planned to research and implement dataset, created in this manuscript, using known models to check the results and compare it to different datasets in later works.

References

1. Sabir M., Banissi E., Child M. Custom Data Augmentation Technique (A Deeper Insight). In: World Conference on Information Systems and Technologies. Cham: Springer International Publishing, 2022. Pp. 75–80.
2. Horro M., J. C. S. P. R., C. H., B. J. Custom High-Performance Vector Code Generation for Data-Specific Sparse Computations. In: Proceedings of the International Conference on Parallel Architectures and Compilation Techniques. 2022. Pp. 160–171.
3. Dembinski H., Olsson F., Spandl J., Sliwa J., Zawadzki R. Custom orthogonal weight functions (COWs) for event classification. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment. 2022. Vol. 1040. P. 167270.
4. Amini M., Sharifani K., Rahmani A. Machine learning model towards evaluating data gathering methods in manufacturing and mechanical engineering. International Journal of Applied Science and Engineering Research. 2023. Vol. 15. Pp. 349–362.

5. Taherdoost H. Data collection methods and tools for research; a step-by-step guide to choose data collection technique for academic and business research projects. International Journal of Academic Research in Management (IJARM). 2021. Vol. 10, no. 1. Pp. 10–38.
6. Ribeiro-Navarrete S., Saura J. R., Palacios-Marqués D. Towards a new era of mass data collection: Assessing pandemic surveillance technologies to preserve user privacy. Technological Forecasting and Social Change. 2021. Vol. 167. P. 120681.
7. Olaniyi O. O., Okunleye O. J., Olabanji S. O. Advancing data-driven decision-making in smart cities through big data analytics: A comprehensive review of existing literature. Current Journal of Applied Science and Technology. 2023. Vol. 42, no. 25. Pp. 10–18.
8. Sun Z., Ma W., Yang Y., Li B., Zhang L. A Novel Efficient Data Gathering Algorithm for Disconnected Sensor Networks Based on Mobile Edge Computing. Wireless Communications and Mobile Computing. 2022. Vol. 2022. P. 4763153.
9. Sharma N. K. Instruments used in the collection of data in research. Available at SSRN 4138751. 2022.
10. Röttger P., Gurevich M., Schutze H., Duran A. Two contrasting data annotation paradigms for subjective NLP tasks. arXiv preprint arXiv:2112.07475. 2021.
11. Greenwald N. F., B. K. A., B. J. W., K. H., G. M., N. T., M. C. Whole-cell segmentation of tissue images with human-level performance using large-scale data annotation and deep learning. Nature Biotechnology. 2022. Vol. 40, no. 4. Pp. 555–565.
12. Wang X., Yu Y., Wu Y., Zhang L., Li Y. A Structure-Guided Molecular Network Strategy for Global Untargeted Metabolomics Data Annotation. Analytical Chemistry. 2023. Vol. 95, no. 31. Pp. 11603–11612.
13. Mamat N., Alzubaidi M., Abdul Hamid S., Ibrahim N., Rahman A. Enhancing image annotation technique of fruit classification using a deep learning approach. Sustainability. 2023. Vol. 15, no. 2. P. 901.
14. Wang S., Cheng X., Zhang H., Shen J., Yu Y. Annotation-efficient deep learning for automatic medical image segmentation. Nature Communications. 2021. Vol. 12, no. 1. P. 5915.
15. Sezen A., Turhan C., Sengul G. A Hybrid Approach for Semantic Image Annotation. IEEE Access. 2021. Vol. 9. Pp. 131977–131994.
16. Reiβ S., O. M., L. H., R. H. Every annotation counts: Multi-label deep supervision for medical image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021. Pp. 9532–9542.
17. Sager C., Janiesch C., Zschech P. A survey of image labelling for computer vision applications. Journal of Business Analytics. 2021. Vol. 4, no. 2. Pp. 91–110.
18. Niu C., Shan H., Wang G. Spice: Semantic pseudo-labeling for image clustering. IEEE Transactions on Image Processing. 2022. Vol. 31. Pp. 7264–7278.
19. Arzt M., Lutz S., H. P., J. H. LABKIT: labeling and segmentation toolkit for big image data. Frontiers in Computer Science. 2022. Vol. 4. P. 777728
20. Roboflow main page. URL: https://roboflow.com/
21. VGG16 Project. URL: https://neurohive.io/en/popular-networks/vgg16/
22. Stanford University project. URL: https://cs231n.stanford.edu/

Tymur Isaiev **Тимур Ісаєв**	Master Student of Computer Engineering & Information Systems Department, Khmelnytskyi National University E-mail: tymuri1112@gmail.com	Магістрант кафедри комп'ютерної інженерії та інформаційних систем, Хмельницький національний університет
Tetiana Kysil **Тетяна Кисіль**	Candidate of Physical and Mathematical Sciences, Associate Professor of Computer Engineering & Information Systems Department, Khmenlnytskyi National University https://orcid.org/0000-0002-4094-3500 e-mail: kysil_tanya@ukr.net	Кандидат фізико-математичних наук, доцент кафедри комп'ютерної інженерії та інформаційних систем, Хмельницький національний університет