SoftwareX 23 (2023) 101504

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

AffecTube — Chrome extension for YouTube video affective annotations

Daniel Kulas, Michal R. Wrobel [*]

Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology, ul. Narutowicza 11/12, Gdansk, 80-233, Poland

ARTICLE INFO

Article history:
Received 30 June 2023
Received in revised form 1 August 2023
Accepted 14 August 2023

Keywords:
Emotion recognition
Dataset
Video annotation

ABSTRACT

The shortage of emotion-annotated video datasets suitable for training and validating machine learning models for facial expression-based emotion recognition stems primarily from the significant effort and cost required for manual annotation. In this paper, we present AffecTube as a comprehensive solution that leverages crowdsourcing to annotate videos directly on the YouTube platform, resulting in ready-to-use emotion-annotated datasets. AffecTube provides a low-resource environment with an intuitive interface and customizable options, making it a versatile tool applicable not only to emotion annotation, but also to various video-based behavioral annotation processes.

Code metadata

Current code version	1.2
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-23-00413
Permanent link to Reproducible Capsule	
Legal Code License	MIT licence
Code versioning system used	git
Software code languages, tools, and services used	Python, JavaScript
Compilation requirements, operating environments & dependencies	Google Chrome browser, docker (optional), Python 3
If available Link to developer documentation/manual	
Support email for questions	michal.wrobel@pg.edu.pl

1. Motivation and significance

The field of deep learning (DL) has made tremendous progress in recent years, revolutionizing fields ranging from image classification to natural language processing. DL models designed for emotion recognition based on facial expressions rely heavily on high-quality annotated image or video datasets as the basis for training and evaluation. These annotations serve as ground truth labels that provide critical guidance to the models. The quantity and quality of these annotations directly influence the performance and accuracy of the DL models [1]. However, the process of annotating large datasets can be a challenging and time-consuming task, often at significant cost. This issue becomes even more pronounced when dealing with video clips, which require manual processing of long footage [2].

The shortage of emotion-annotated video datasets for supervised learning is mainly due to the significant costs associated with the manual annotation process [3]. The annotation process requires extensive expertise, including familiarity with the taxonomy used, the tools, and the nuances of emotion recognition itself. This combination of expertise and effort contributes to the limited availability of such datasets [4].

Overcoming the challenges associated with the annotation process is critical to the advancement of DL models for emotion recognition. Efficient and cost-effective annotation methods are needed to overcome the time and cost constraints that often accompany the annotation of large datasets. By streamlining the annotation workflow, researchers and practitioners can improve the scalability and applicability of DL models, opening up new possibilities for understanding human emotion. These advances have far-reaching implications in diverse domains such as affective computing, human–computer interaction, and social robotics. Optimizing the annotation process is a critical step in fostering continued progress and innovation in DL models tailored for emotion recognition based on facial expressions.

One possible approach to address the challenges of the emotion annotation process in large video datasets is to use the

[*] Corresponding author.
E-mail address: michal.wrobel@pg.edu.pl (Michal R. Wrobel).

https://doi.org/10.1016/j.softx.2023.101504

crowdsourcing technique. By harnessing the power of crowds, these techniques distribute the annotation workload across a large group of contributors, enabling the annotation of large datasets in a faster and more cost-effective manner. Furthermore, crowdsourcing techniques provide an opportunity for global participation, allowing contributors from different cultural backgrounds and demographics to annotate the datasets. This diversity can enhance the robustness and generalizability of DL models, ensuring that they perform well across different populations and contexts.

However, it is important to address potential challenges associated with crowdsourcing, such as maintaining annotation quality and dealing with potential biases. Robust quality control mechanisms, clear annotation guidelines, and iterative feedback loops can help mitigate these concerns.

Many of the publicly available emotion-tagged video datasets have been created using either posed emotions or self-reported participant data [5]. In the case of posed emotions, actors are instructed to display specific emotional expressions, which may not fully capture genuine emotions experienced in real-world scenarios. Self-reporting involves participants indicating their emotional states while watching videos, which can be subjective and influenced by various factors. While these approaches are valuable for certain research purposes, they do not always reflect the complexity and variety of emotions experienced in naturalistic settings.

In the process of emotion annotating videos recorded in natural (''in the wild'') environments, researchers often develop specialized computer software designed specifically for their studies [6–9]. These software tools are tailored to support the precise needs of the research. On the other hand, generic behavioral event logging programs like BORIS are rarely used for this purpose [10].

In addition to the use of dedicated software tools, there have been advances in the development and sharing of tools that facilitate the process of crowdsourcing annotation. One example is PAGAN, a web application that provides a platform for organizing and conducting annotation research [11]. PAGAN allows researchers to outsource annotation tasks by simply sharing a link to the service, enabling efficient collection of annotations from a variety of contributors. Another tool, Covfee, is specifically designed to support continuous annotation and offers crowdsourcing capabilities [12]. However, deploying such programs necessitates their installation on a server equipped with ample resources. For example, annotating large video files may require a significant amount of disk space and a broadband Internet connection. This requirement arises from the need to store and serve a considerable volume of video files. Furthermore, the installation process itself may entail a higher level of system administration proficiency. Configuring and setting up the software often calls for advanced knowledge and expertise in system administration to ensure a smooth and successful installation.

To address these challenges, we developed AffectTube, an extension for the Chrome web browser that facilitates a low-resource environment for annotating video footage directly on the YouTube platform. To annotate the collected video files, they are first uploaded to the YouTube platform. Then, with the extension installed, annotators can seamlessly start the annotation process directly on the YouTube website.

An important advantage of this approach is the ability to annotate not only independently collected video footage, but also any available YouTube resource. This provides a convenient means of annotating videos from extensive multimedia library, which is the largest repository of online video today. However, it is important to note that in such cases, careful attention should be paid to the licensing terms under which the videos are published.

The relevance of AffecTube lies in its ability to significantly reduce the setup time for video annotation while operating within a resource-constrained framework. By leveraging the YouTube platform, it provides a streamlined and efficient solution for annotating videos, making it accessible to a wider range of scientists and minimizing the hardware requirements typically associated with video annotation systems.

2. Software description

The core component of the presented software is the AffecTube extension for annotating videos directly on the Youtube platform page in the Chrome web browser. The process of preparing the environment for crowdsourced annotation consists of only four steps. The first is to prepare the videos to be annotated and upload them to the YouTube platform. Alternatively, videos can be selected from the entire YouTube library. The second step is to define the list of labels that will be available to annotators in the extension's source file. The next step is to specify the repository where the results of the annotators' work will be stored. It is possible to use the GitHub platform or own server. The appropriate configuration should also be set in the source file. The last step is to send the extension to the annotators in the form of an archive, together with the user manual and the annotation protocol.

To facilitate the dataset creation process, two additional components have been developed. The first is a lightweight API server designed to store the annotation results. It takes advantage of containerization to provide a portable way to deploy in different environments. The aim of the second component is to retrieve YouTube video fragments that have been annotated as containing emotions.

2.1. Software architecture

Fig. 1 shows the architecture of the developed software based on the three components, namely ''Chrome extension'', ''Storage API'' and ''Dataset creator''. The Storage API component is optional, as it can be replaced by storing the annotation results in a GitHub service.

Once videos are uploaded to the YouTube platform, annotators begin labeling the emotions in the videos they watch. The footage is displayed on the YouTube website and labels are assigned using the AffecTube extension. Once completed, the results are sent to the storage service in a single PUT HTTP request method. Depending on the configuration, they end up either on the dedicated server or in a GitHub repository, where they are stored in a JSON file.

The next step, after all the annotators have finished labeling, is to process the collected data and eventually create a dataset. To accomplish this, it is necessary to transfer all the collected JSON files and run the Dataset Creator script. As a result, video files with fragments labeled with emotions and a JSON file with a summary of the content of the dataset are produced.

The AffecTube extension was developed in vanilla JavaScript, in accordance with Google's documentation for Chrome extension developers. The next component, ''Storage API'', which provides a simple REST service to allow the storage of data from the extension, was developed in Python using the *FastAPI* web framework. Finally, the ''Dataset creator'' script was also written in Python, using the *pytube* library to download videos from YouTube and *ffmpeg-python* library to extract labeled video fragments.

2.2. Software functionalities

The following sections describe the functionality of each component.

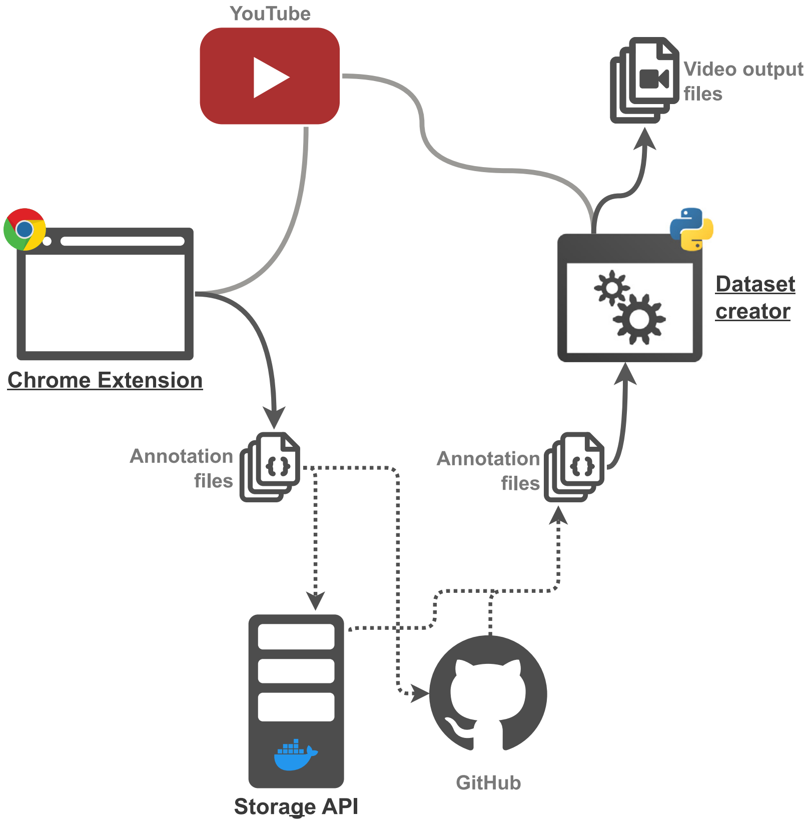


Fig. 1. Components of the AffecTube.

2.2.1. *AffecTube extension*

The primary purpose of the extension is to allow annotators to easily and intuitively label emotional states directly on the YouTube page. After installing the extension, additional buttons will be displayed on each YouTube video according to the labels defined in the source code. Additionally, the user can send the data to the storage service using the extension's pop-up window once the video labeling process is complete.

2.2.2. *Storage API*

A simple REST server, the Storage API, provides an endpoint to which the extension sends data in JSON format from the annotation process. The received data is stored in a text file. Only a PUT endpoint is provided by the service.

2.2.3. *Dataset creator*

The purpose of the script is to process annotation data collected in JSON files and generate emotion-labeled video fragments. The configuration file and command line options allow customization of the script's execution. The following options can be defined in this way:

- Format of the video output files.
- Acceptance threshold for the number of annotators who have annotated a fragment.
- Ignore label matches, useful for initial annotation using crowdsourcing technique, where the resulting fragments will be labeled by experts in a further step.
- Margin size for trimmed video fragments.
- The maximum size of the annotated fragment to process, useful for filtering out negligent annotators.
- Input and output directories and files.

3. Illustrative example

AffecTube supports the process of creating emotion labeled video datasets from recorded footage. Due to its extensive configuration capabilities, it can be used to perform both initial video filtering using crowdsourcing techniques and to prepare a ready-to-use dataset based on expert labeling.

The following illustrative example is based on a use case in the development of an extended version of the DevEmo dataset, which contains recordings of students' facial expressions while solving programming tasks [10]. Within the previous studies [13, 14] over 1000 h of video footage was collected, with only a small portion already manually labeled. Since labeling such an extensive is extremely costly and time-consuming, it was decided to use crowdsourcing techniques.

Before the labeling process began, the fragments most likely to contain facial expressions were selected, resulting in 16, 20 min video clips. As AffecTube allows labeling of footage available on YouTube, the prepared clips were uploaded to the platform. To limit outside access to the footage, the option to not show on YouTube listings has been selected. Alternatively, accessibility can be set to private mode. In this case, each annotator must be manually granted access to the footage. In addition, it is possible to label any recording available in YouTube's vast library. In both cases, it is enough to simply provide the annotators with a link to the video on the YouTube platform to be labeled.

Then a list of available labels has been defined directly in the AffecTube extension code in variable *emotions* in the main function of the *src/inject/inject.js* file. In this way, it is possible to define any list of labels that will be available to annotators. For the case study presented here, the following labels were chosen "happiness", "sadness", "disgust", "fear", "surprise", "anger" and "confusion", as shown in Listing 1.

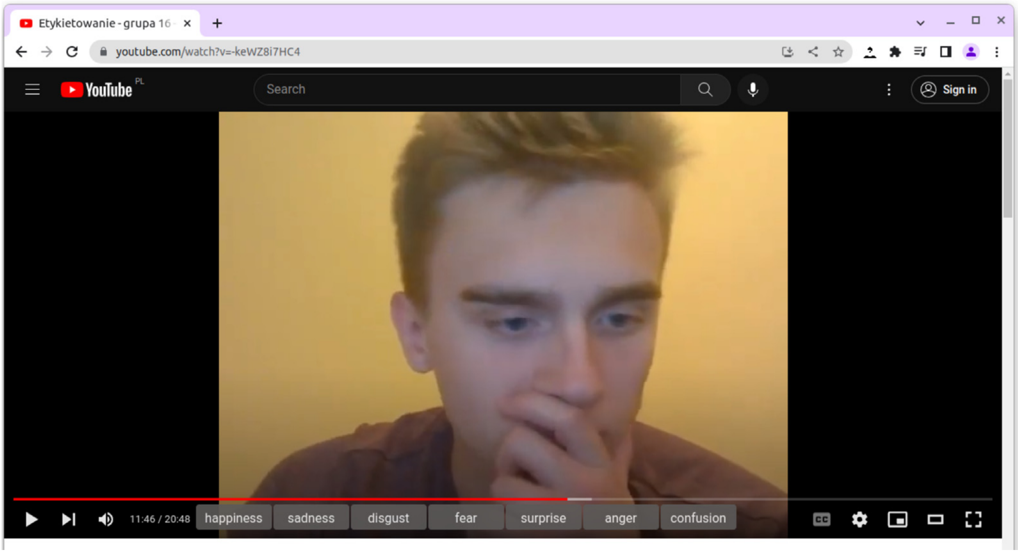


Fig. 2. Screenshot of the AffecTube Chrome extension.

Listing 1: Definition of labels in src/inject/inject.js file

```
...
(() => {
  let emotions = ["happiness", "sadness", "disgust",
                  "fear", "surprise", "anger",
                  "confusion"];
  let emotionsButtonStates = Array.apply(
        null, Array(emotions.length)).map(e => -1
  );
  let savedLabels = [];
  let videoPlayer;
  let videoURL;
...
```

The extension does not currently support the ability to label more than one object with the same labels. However, it is possible to work around this issue by defining repeated labels for different objects, e.g. *S1 smiling*, *S1 crying*, *S2 smiling*, *S2 crying*.

Students at the Technical University of Gdansk were invited to participate in the study, and the final 203 volunteers were divided into 16 groups, each assigned a different video to label. Each participant was provided with the source code of the extension, a link to the video, and an annotation protocol.

Before starting the annotation process, each participant installed the received extension in the Chrome browser. In *Extensions/Manage Extensions* in the settings it was required to activate the *Developer mode* in the upper right corner. Then, by pressing the *Load unpacked* button, the directory containing the extension had to be pointed to. At this point the extension was installed and ready for labeling.

After opening the YouTube page from the received link, the defined labels were displayed at the bottom of the window, as shown in Fig. 2. Annotation is performed during playback by clicking the label button at the beginning of the expression. In this case, the background of the clicked label changes to dark red. A second click on the same label ends the fragment with the expression. By marking the beginning of two or more fragments before the end of others, it is possible to annotate overlapping fragments with different labels.

Users could use keyboard shortcuts available on the YouTube platform to select fragments more precisely, such as the space bar to pause playback, or the ,. keys to move forward or backward one frame.

The extension also allows modification of the selected fragments in the pop-up window. Clicking on the extension icon in the upper right corner of the browser, will open the window shown in Fig. 3. All selected fragments with assigned labels are displayed in the table. Individual labels can be removed or their start and end times can be changed. In addition, pressing the arrow icon next to the time will rewind the video to the specified fragment.

After annotating the video, participants sent the data to a storage server by entering the annotator code in the *Username* field and pressing the *Submit* button in the extension pop-up window, showed in Fig. 3. The data sent by each annotator was stored by the Storage API in a separate JSON file.

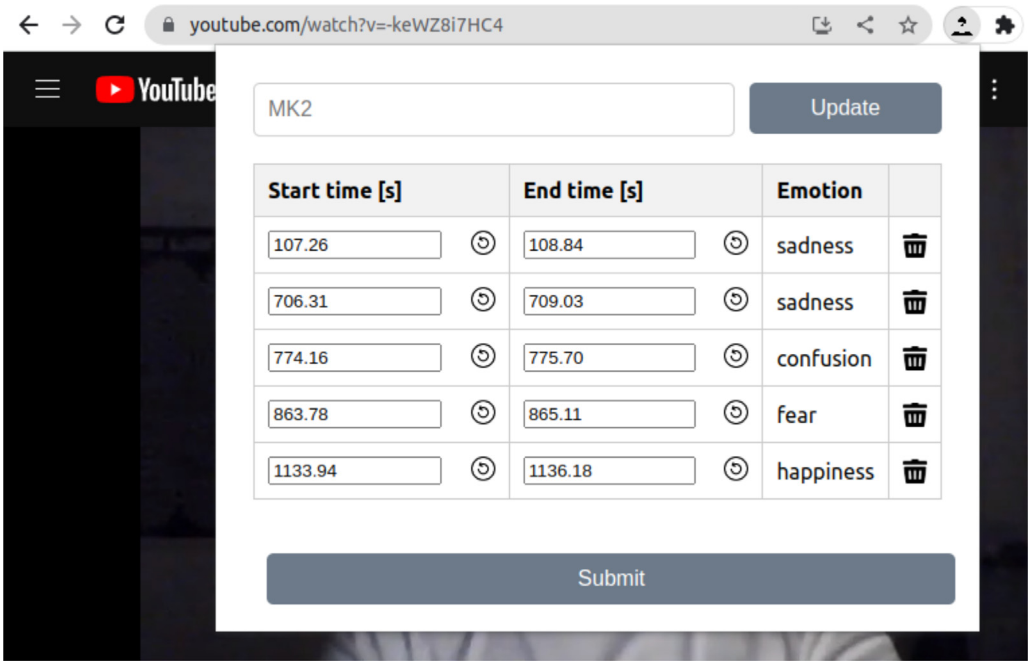
After all participants completed the annotation process, the JSON files containing the data were transferred to a local computer, where video fragments with facial expressions were extracted using the *Dataset creator* script.

The functionality of the script can be modified by making changes to the *config.json* configuration file, as shown in Listing 2, or by adding options when run from the command line.

Listing 2: Dataset creator configuration file

```
{
  "input_dir": "../annotations",
  "output_dir": "./dataset",
  "output_filename": "videos_annotations.json",
  "acceptance_threshold": 0.5,
  "match_labels": false,
  "raw_annotations_only": false,
  "merged_annotations_only": false,
  "download_from_youtube": true,
  "max_fragment_size": 30,
  "margin": 4
}
```

This way it is possible to define the location of the annotated input JSON file directory (*input_dir*), the output directory where the generated fragments will be saved (*output_dir*), or the name of the file where the metadata describing the dataset will be stored (*output_filename*). In addition, *acceptance_threshold* can be specified, which is defined as the ratio of annotators who have labeled a particular fragment to the number of all video annotators. In this way, the level of reliability of the annotation can be controlled.



MK2

Start time [s]	End time [s]	Emotion	
107.26	108.84	sadness	
706.31	709.03	sadness	
774.16	775.70	confusion	
863.78	865.11	fear	
1133.94	1136.18	happiness	

Fig. 3. Screenshot of the AffecTube Chrome extension's popup.

The *match_labels* option can be used to specify whether matches of assigned labels should be taken into account. When using crowdsourcing, it is possible to generate fragments that have been indicated by the appropriate number of annotators without checking for label matches. This will result in a collection of recordings that contain some kind of facial expression, and it is only in the next phase that experts will assign the accurate labels.

The next two options, *raw_annotations_only* and *merged_annotations_only*, allow intermediate data from the annotation analysis and merging process to be saved, which can be useful for manual data processing. The *download_from_youtube* option makes it possible to specify whether the videos are to be downloaded from YouTube or whether they have already been placed in the output directory. This way it is possible to reduce unnecessary transfer of data that was downloaded during the previous execution of the program.

In order to eliminate careless annotators, the maximum indicated duration of emotion (*max_fragment_size*) can be determined. In the study conducted, it was concluded that some participants did not act diligently and marked fragments of several minutes as those containing expressions of emotion. Finally, with the *margin* option, it can be defined in seconds the margin that will be added to the generated fragments, before the moment of the beginning of the expression of emotion and after the end.

The final step is to run the script. Listing 3 shows a portion of the output of the script run with the settings shown in Listing 2.

Listing 3: Dataset creator run

```
Downloading https://youtu.be/Cu7pKk8KpOY ...
Video downloaded: ./dataset/Cu7pKk8KpOY.mp4
Processing Cu7pKk8KpOY.mp4, extracting 5 fragments
Extracting: ./dataset/Cu7pKk8KpOY_1.mp4
Extracting: ./dataset/Cu7pKk8KpOY_2.mp4
Extracting: ./dataset/Cu7pKk8KpOY_3.mp4
Extracting: ./dataset/Cu7pKk8KpOY_4.mp4
Extracting: ./dataset/Cu7pKk8KpOY_5.mp4
Downloading https://youtu.be/8ZUQGOUOO_M ...
Video downloaded: ./dataset/8ZUQGOUOO_M.mp4
Processing 8ZUQGOUOO_M.mp4, extracting 4 fragments
Extracting: ./dataset/8ZUQGOUOO_M_1.mp4
Extracting: ./dataset/8ZUQGOUOO_M_2.mp4
Extracting: ./dataset/8ZUQGOUOO_M_3.mp4
Extracting: ./dataset/8ZUQGOUOO_M_4.mp4
...
```

As a result of the script, 16 videos were downloaded from YouTube and, for each, an average of five fragments were extracted that were explicitly identified by annotators as those containing emotions and saved in output directory. The script also produced a JSON file that contains metadata describing the created dataset. The beginning of the file generated in the previous step is shown in Listing 4.

Listing 4: Dataset metadata

```
[
    {
        "video_code": "Cu7pKk8KpOY",
        "fragments_count": 5,
        "annotations": [
            {
                "startTime": 72.31,
                "endTime": 100.63,
                "annotators_count": 7,
                "annotators": [
                    "192594",
                    "193088",
                    "193437",
                    "191527",
                    "193251",
                    "193276",
                    "193176"
                ],
                "labels": [
                    "sadness"
                ],
                "agreement_ratio": 0.7,
                "filename": "Cu7pKk8KpOY_1.mp4"
            },
            ...
```

The file contains a list of objects for each annotated video file. The *video_code* field contains information about the video's YouTube identifier, and *fragments_count* contains the number of identified fragments that meet the acceptance threshold specified in the configuration. Next, *annotations* contains a list of objects that describe a single fragment. They contain information about the start and end time of the fragment, the number of annotators who indicated it, a list with the codes of these annotators, a list with the labels assigned to the fragment, the achieved agreement ratio, and finally the name of the video file in which the fragment was saved. This file can be the basis for preparing a more comprehensive description of the dataset, but is also sufficient as it is for most cases.

Another way to use the presented software is to use the crowdsourcing technique as a stage of input data reduction, i.e. to reduce the length of the video that is manually annotated by experts. In this case, it is possible to ignore the matching of the labels assigned by the annotators by setting the *match_labels* option to false. The metadata file will then contain a list of all labels assigned to the fragment. In the case presented here, the list of generated fragments is doubled, when setting *match_labels* is set to false. These fragments can then be annotated by experts also using the AffecTube extension.

At the end of the use case described above, participants were asked to complete a short satisfaction survey. Of the 119 responses collected, 68.9% agreed that the solution provided was intuitive and easy to use. It is noteworthy that only 3 people encountered difficulties while completing the assigned labeling task. These results may serve as confirmation that integrating video annotation capabilities into the well-known interface of the YouTube platform significantly reduces the learning curve associated with adopting a new tool.

4. Impact and sustainability

AffecTube provides a cost-effective solution that requires minimal resources to annotate facial expression video files with emotional labels. By leveraging the YouTube platform, the extension enables the utilization of crowdsourcing techniques, tapping into a vast pool of annotators who are already familiar with the interface. This approach significantly reduces the overall costs associated with the annotation process, both on the researcher's side and globally. The reliance on existing YouTube infrastructure eliminates the need for additional servers to store and serve large volumes of video content, thereby saving substantial resources and infrastructure expenses. In addition, the developed script allows seamless creation of a ready-to-use video dataset with a metadata description. This minimizes the effort required to manually process the annotation results.

Beyond its application in emotion annotation, AffecTube exhibits versatility as a valuable tool for various behavioral annotation processes. By allowing the flexibility to modify and customize labels, the extension can be adapted to annotate a wide range of behavioral aspects, such as gestures, actions, and interactions.

The impact of AffecTube extends to researchers and developers in the fields of emotion analysis and machine learning. With the increased availability of emotion-annotated video datasets, supervised learning models can benefit from more diverse and representative training data. This can lead to improved accuracy and robustness of emotion recognition systems, enabling advances in various applications such as affective computing, human–computer interaction, and multimedia content analysis.

5. Conclusions

This paper presents AffecTube software for preparing datasets of emotionally annotated video clips. The ability of the Chrome browser extension to seamlessly integrate with the YouTube platform allows annotators to use the familiar interface and functionality to annotate behavioral cues directly in the context of video content. The developed script allows to automatically generate a ready-to-use dataset of emotionally annotated video fragments based on collected data from annotators using YouTube resources.

By addressing the challenges associated with the shortage of emotion-annotated video datasets and the costs of manual annotation, AffecTube provides a cost-effective and accessible solution through crowd-based annotation approach and integration with the YouTube platform. Its low-resource environment, user-friendly interface, and customization options make it a versatile tool that can be applied not only to emotion annotation but also to various behavioral annotation processes. The impact of AffecTube extends to researchers, annotators, and the global research community, enabling advancements in emotion recognition, affective computing, and related fields.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] Mariooryad S, Busso C. Analysis and compensation of the reaction lag of evaluators in continuous emotional annotations. In: 2013 humaine association conference on affective computing and intelligent interaction. IEEE; 2013, p. 85–90. http://dx.doi.org/10.1109/ACII.2013.21.

[2] Gera D, Balasubramanian S. Noisy annotations robust consensual collaborative affect expression recognition. In: Proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 3585–92. http://dx.doi.org/10.1109/ICCVW54120.2021.00399.

[3] Liu Y, Xie X. An annotator solves the cold-start problem by using emotional label replacement and consistent selection mechanism. In: 2022 international joint conference on neural networks (IJCNN). IEEE; 2022, p. 1–8. http://dx.doi.org/10.1109/IJCNN55064.2022.9892830.

[4] Böhm T, Engel F, Bzdok D, Schneider F, Hemmje M. Approach to semi-automatic labeling of video sequences for affective computing-enabling the comprehensive assessment of emotion detection software from mimics. In: 2017 IEEE international conference on bioinformatics and biomedicine (BIBM). IEEE; 2017, p. 1996–2000. http://dx.doi.org/10.1109/BIBM.2017.8217967.

[5] Li W, Cui Y, Ma Y, Chen X, Li G, Zeng G, et al. A spontaneous driver emotion facial expression (defe) dataset for intelligent vehicles: Emotions triggered by video-audio clips in driving scenarios. IEEE Trans Affect Comput 2021. http://dx.doi.org/10.1109/TAFFC.2021.3063387.

[6] Barros P, Churamani N, Lakomkin E, Siqueira H, Sutherland A, Wermter S. The OMG-emotion behavior dataset. In: 2018 international joint conference on neural networks (IJCNN). IEEE; 2018, p. 1–7. http://dx.doi.org/10.1109/IJCNN.2018.8489099.

[7] Khanh TLB, Kim S-H, Lee G, Yang H-J, Baek E-T. Korean video dataset for emotion recognition in the wild. Multimedia Tools Appl 2021;80:9479–92. http://dx.doi.org/10.1007/s11042-020-10106-1.

[8] Cao H, Cooper DG, Keutmann MK, Gur RC, Nenkova A, Verma R. Crema-d: Crowd-sourced emotional multimodal actors dataset. IEEE Trans Affective Comput 2014;5(4):377–90. http://dx.doi.org/10.1109/TAFFC.2014.2336244.

[9] Shen G, Wang X, Duan X, Li H, Zhu W. Memor: A dataset for multimodal emotion reasoning in videos. In: Proceedings of the 28th ACM international conference on multimedia. 2020, p. 493–502. http://dx.doi.org/10.1145/3394171.3413909.

[10] Manikowska M, Sadowski D, Sowinski A, Wrobel MR. DevEmo—Software developers' facial expression dataset. Appl Sci 2023;13(6):3839. http://dx.doi.org/10.3390/app13063839.

[11] Melhart D, Liapis A, Yannakakis GN. PAGAN: Video affect annotation made easy. In: 2019 8th international conference on affective computing and intelligent interaction (ACII). IEEE; 2019, p. 130–6. http://dx.doi.org/10.1109/ACIIW.2019.8925149.

[12] Quiros JV, Tan S, Raman C, Cabrera-Quiros L, Hung H. Covfee: an extensible web framework for continuous-time annotation of human behavior. In: Understanding social behavior in dyadic and small group interactions. PMLR; 2022, p. 265–93.

[13] Wrobel MR. Applicability of emotion recognition and induction methods to study the behavior of programmers. Appl Sci 2018;8(3):323. http://dx.doi.org/10.3390/app8030323.

[14] Wrobel MR, Zielke AW. Maliciouside-software development environment that evokes emotions. In: 2018 federated conference on computer science and information systems (FedCSIS). IEEE; 2018, p. 1009–12. http://dx.doi.org/10.15439/2018f311.