ARTIFICIAL INTELLIGENCE-BASED EMOTION RECOGNITION SYSTEMS TO ENHANCE STUDENT MOTIVATION

J.M. Bravo-Pacheco, I. Martín-Fernández, S. Esteban-Romero, M. Gil-Martín, F. Fernández-Martínez

Grupo de Innovación Docente en Ingeniería y Sistemas Electrónicos (GRIDS), Departamento de Ingeniería Electrónica, Universidad Politécnica de Madrid (SPAIN) (SPAIN)

Abstract

In contemporary education, traditional methodologies often struggle to captivate today's learners, who are deeply immersed in a fast-paced, tech-driven world. To effectively engage this generation, it's imperative to adopt innovative teaching methods that cater to diverse learning styles and leverage cutting-edge technologies. This challenge is particularly pertinent in developing an emotion recognition system capable of analyzing raw images or facial landmarks to discern the emotions exhibited by individuals.

Our Educational Innovation Project, "DEMOSEI - Design and implementation of demonstrator systems for their application in the teaching-learning process in Intelligent Electronic Systems subjects," aims to pioneer a novel approach in integrating emotion recognition technology into educational settings. This work focuses on utilizing deep learning models, such as Convolutional Neural Networks (CNNs) and facial landmark detection algorithms, to accurately identify and interpret human emotions. By harnessing the power of these artificial intelligence-driven tools, students can explore and understand the intricacies of emotional cues through practical applications.

The project encompasses a comprehensive pipeline from data acquisition and preprocessing to model training and deployment, empowering students to delve into every stage of artificial intelligence development. The system is designed as an open platform, encouraging students to experiment with different deep learning architectures and customization options. Leveraging TensorFlow and Keras in Python, the system supports deployment on versatile platforms like Google Collab or Raspberry Pi, ensuring accessibility and scalability in educational environments.

The system holds significant relevance across diverse academic disciplines, including computer science, psychology, and artificial intelligence. Its interdisciplinary nature not only enhances educational experiences but also fosters collaboration and innovation among students from varied backgrounds. By bridging the gap between theoretical knowledge and practical application, this system prepares learners to navigate the evolving landscape of artificial intelligence and contribute meaningfully to the field.

Keywords: Deep learning, emotion recognition, challenge-based learning methodology, student motivation.

# INTRODUCTION [Arial, 12-point, bold, upper case and left alig.]

The Educational Innovation Project “DIRASEI”, where this work is framed, is focused on designing the most appropriate contents for subjects with a practical component in order to contribute to the improvement of student learning outcomes and motivation. This way, the generated resources are based on innovative technology and could be used in subjects that follow a challenge-based learning methodology. These resources could be crucial to increase the students’ motivation, which is important for their academic achievement and enjoyment [1].

To complete that objective, the generated resources are focused on Facial Emotion Recognition (FER). These systems are concerned with identifying and interpreting human emotions based on facial expressions. It involves analyzing facial features such as eyes, mouth, and facial muscles to determine emotions like happiness, sadness, anger, surprise, etc. These systems have wide application fields such as education (tracking the attention of the students), public safety (lie detectors), healthcare (depression or autism detection) or employment (helping recruiters).

This work describes an intelligent electronic system proposed to enhance the students’ curiosity and offer them artificial intelligence-based resources for their ongoing education. This system is based on recognizing human emotion from an image of a face. In addition, this work describes the response from a small sample of students that used the proposed resources.

# ABOUT THE USE OF INFORMATION TECHNOLOGIES AND INNOVATIVE TEACHING TECHNIQUES

In the GRIDS educational innovation group (Grupo de Innovación Docente en Ingeniería y Sistemas Electrónicos, [link](https://innovacioneducativa.upm.es/informacion-grupo?grupo=187)) we make extensive use of different technologies in teaching, along with innovative teaching techniques. In all subjects, we utilize the Moodle platform [6] as an essential resource for various purposes: distributing/gathering information, submitting/evaluating tests/assignments, continuous assessment, bidirectional communication between teachers and students, FAQs, forums where students pose questions for teachers and peers to answer, and more. Similarly, we have thematic channels on YouTube through which we provide explanatory videos, demonstrations, and, most importantly, showcase student work to inspire and nurture students' curiosity. We integrate virtualization and emulation tools (e.g., VirtualBox), rely on cloud platforms for practical projects (e.g., Google Colab), use them as a backup for student work (e.g., OwnCloud), provide access to other freely available online educational resources (e.g., AWS, MSDN, etc.), and use them as version control and repository tools for teamwork (e.g., GitHub), among other uses.

Next, we will detail some of our pedagogical approaches together with the specific use of various innovative techniques:

* **Service Learning**: Within the context of the SDGII course, a pivotal component of our Telecommunication Technologies and Services engineering program at UPM, a unique initiative known as "Innovative Projects" has been developed. This initiative challenges students to actively engage in special projects designed to benefit the community. Within this framework, our instructional team not only imparts essential course concepts but also guides students in understanding the genuine needs of end-users, enabling them to make tangible and effective improvements to their surroundings. This emphasis on real-world application is particularly prominent during the project's design phase. The initiative's most recent edition saw the successful completion of five projects, each exemplifying our commitment to service learning. Notably, three of these projects focused on monitoring environmental conditions, with a specific emphasis on air quality, across diverse settings such as hospitals, classrooms, and gymnasiums. Another remarkable project involved the development of specialized augmented reality glasses equipped with an audio description service tailored for the deaf and mute community. Finally, the fifth project aimed to streamline evaluation and administrative procedures within the academic realm. For a closer look at these innovative systems, please visit the following [link](https://youtube.com/playlist?list=PL_MyZ0HUbo2BytQBGTRQ2pIzu3GW5qZW4).
* **Design Thinking**: In the MCHP course, which is held annually and covers methodology, quality, and personal skills, newly enrolled students in the UPM Master's Degree in Electronic Systems Engineering (MUISE) begin with a thematic module focused on innovation. Here, they acquire essential competencies related to understanding the significance, processes, and tools of innovation. Subsequently, they apply these competencies to create innovative projects. Professors from the Electronic Systems Laboratory (LSEL), a course in the second semester, collaborate during this initial module, promoting creativity and innovation through the Design Thinking methodology. Students are presented with technological challenges that they must address in groups, applying engineering project development and management methodologies. They form working groups, go through the initial stages of the Design Thinking process, and eventually create an initial prototype for their proposed solution. Professors from LSEL participate in evaluating the solutions proposed at this stage to ensure their compatibility with the embedded systems technologies available in the course. The implementation of solutions and the iterative evaluation phases within the Design Thinking framework occur in the LSEL course during the second semester.
* **Challenge-Based Learning**: At INSE, substantial efforts are underway to integrate this methodology into the curriculum. Specifically, for a course that concentrates on developing machine learning solutions optimized for deployment in cost-effective electronic systems with minimal energy consumption and computational resources, the approach of utilizing real and practical cases is essential. This scenario corresponds to the systems under description and analysis in the current study. In recent years, additional use cases have been introduced into the course to enrich the learning experience of students. Inspired by these use cases, the members of the GRIDS  team have been developing and expanding a family of demonstrators used to carry out various automated tasks. These demonstrators are designed and implemented using different combinations of microcontrollers and types of sensors, such as inertial sensors, cameras, and other affordable electronic components. Here, we enumerate some of the most interesting examples of these demonstrators:
* Object recognition employing a camera ([link](https://youtu.be/HSGZzN4CJDo)).
* Detection of anomalies in data processing centers.
* Analysis of sentiment in social media comments ([link](https://youtu.be/IJxiwOdDG80)).
* Semantic image segmentation for autonomous navigation ([link](https://youtu.be/b8ISQyqsZvM)).
* Recognition of gestures and physical activities based on inertial sensors and recognition of the human body poses for behavior modeling (link).
* Recognition of emotions expressed by humans based on facial expression analysis. (That is detailed in the subsequent sections)

These practical cases help students better identify areas where machine learning can be successfully applied, recognize representative real-world problems that it can solve, and serve as a source of ideas that can be highly beneficial in their professional careers.

* **Collective Intelligence**: In the academic context, collective intelligence is cultivated through teamwork. Agile methodologies, like those implemented in LSEL, place a significant emphasis on team management and enable the full utilization of collective intelligence. These methodologies deliberately aim to encourage the emergence of intelligent behaviors resulting from interactions among individuals. By embracing them, instructors place their trust in the team, the experience and skills of its members, and their decision-making capacity, thereby promoting the team's development towards self-management and harnessing collective intelligence as a means to successfully accomplish a project.
* **Adaptive Learning**: Adaptive learning is a particularly valuable methodology in courses like SDGII. In this course, sessions are structured around milestones related to various versions of the prototype to be implemented, each corresponding to different levels of development or maturity (functionality) achieved by the prototype. The functional requirements for each version are presented to students beforehand, with careful consideration to ensure that these requirements are clear and easily verifiable by the students themselves. Utilizing Moodle and its features, such as quizzes and submission boxes, we gather comprehensive data on self-assessment provided by students on a weekly basis regarding their progress against the planned schedule. Self-assessment of performance is crucial for students as it offers immediate feedback, enabling them to adapt their efforts based on their needs, errors, and successes. As for the teaching team, the analysis of the extensive data collected allows us to continuously and promptly adjust the pace and scope of the different sessions, tailoring them to any needs, issues, or potential delays that may arise at any given moment. This adaptation occurs on three distinct levels: weekly at the group or work shift level (there are five different shifts in the course, one for each day of the week; the theoretical introduction at the beginning of each session is adjusted based on observed circumstances); weekly at the individual level through in-class personal attention; and finally, also at the individual level but coinciding with mid-term assessments. In all cases, we strive to adapt the course's development as swiftly as possible to address any difficulties or areas of improvement that may affect the students, thereby encouraging them to become more active participants in their own learning.

# METHODOLOGY

The generated systems contain different modules in order to address each step of a project lifecycle, including data acquisition, image processing, training and analysis of the results, and deployment. Fig. 1 shows the general architecture for the generated resources.

*Imagen que contiene Diagrama

Descripción generada automáticamente*

*Figure 1. General architecture for the generated resources.*

The intelligent electronic system proposed in this work is focused on FER. Specifically, the system is capable of detecting the facial emotion that the user is expressing in real time. The system is developed into different stages. First, the data acquisition is performed. Afterwards, since the system relies on landmark extraction for classification, the images are pre-processed to extract 478 points, which are the features inserted into the model, as it will be explained in the *Image processing* section. Then an artificial intelligence-based system is trained for the emotion recognition, and, at the end, the model is deployed for real-time testing.

For these objectives, the recording and deployment stages are performed in a Raspberry Pi module, while the training process is made with Google Collab. By using this combination, we allow for the students to experiment with low resource devices for the less demanding operations, and with cloud computing for the heavy ones. Students will also practice using Python libraries for data acquisition, image processing such as MediaPipe, NumPy and OpenCV, and machine learning such as TensorFlow and Keras.

## Data acquisition

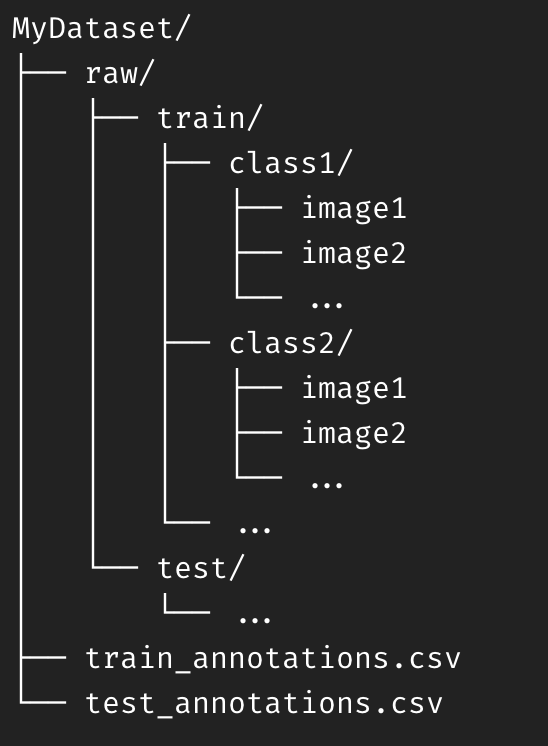
The project allows the students to work both with existing datasets found on the internet or custom ones recorded by themselves. In this sense, the students can test the changes between using large datasets, which are usually less accurate to the desired functionality, small datasets, usually more focused on the task, or even combining them, leading to more robust systems.

The RAVDESS dataset is proposed as a baseline. It contains 24 professional actors (12 female, 12 male), vocalizing two lexically-matched statements in a neutral North American accent. The dataset also contains the videos of the actors saying their phrases. Speech includes seven emotions: calm, happy, sad, angry, fearful, surprise, and disgust. Each expression is produced at two levels of emotional intensity (normal or strong), with an additional neutral expression. It also contains samples of the actors singing, which were not used. Finally, as the dataset is composed of videos, the frames were extracted, leading to images where the emotion does not correspond exactly to its label.

On the other hand, the students are allowed to record their own dataset for training the systems. In this sense, they can decide the number of classes to record, their class names and the number of images to save at each class. This personalization of the process allows them to test different combinations and to see their effect in the model.

The recording process is done with a Raspberry Pi with a camera module connected to it. This process is done with a Python script and using the picamera module to configure the camera. The process integrates the following steps:

1. **Configuration of recording settings**. As mentioned before, the student can configure different settings of the data acquisition process. Specifically, they can configure the recording resolution, the classes to record, the number of images to capture per class and the distribution between train and test subsets.
2. **Image capturing.** After setting those parameters, the script is ready to be run. A window will open showing the camera preview and indicating the class that is going to be recorded. This preview does also display 478 face landmarks, indicating that the face is being detected. More about these landmarks is explained in the *Image Processing* section. Once the user presses the (s)tart key, the script will capture the number of images specified every 0.5 s.
3. **Image saving.** After the images are recorded, they are saved into a directory with all the datasets. At the same time, an annotations file is generated for later use of the images. This file will contain the label of each image and their path. The structure of the directory can be seen in Fig. 2.



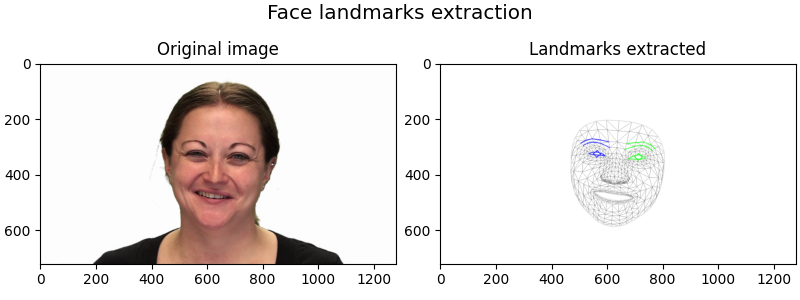
*Figure 2. Directory structure for the recorded dataset.*

## Image Processing

The images saved need to be loaded and preprocessed before inserting them into the model. For doing this, a data pipeline is proposed. This pipeline is designed to be easily modified, allowing for the student personalization, and fast in processing the images. Inside the data flow, the following steps are done:

### Landmark extraction

Looking for an optimization of the system, the project proposes the use of 478 landmarks extracted from the face to perform the classification. These landmarks are extracted via the MediaPipe Face library, published by Google. Specifically, the utility takes an image in RGB format and returns an array with 478 points, represented as x, y, z coordinates. These values are also normalized between 0 and 1 with respect to the size of the input image. The landmark extraction process can be seen in Fig. 3.

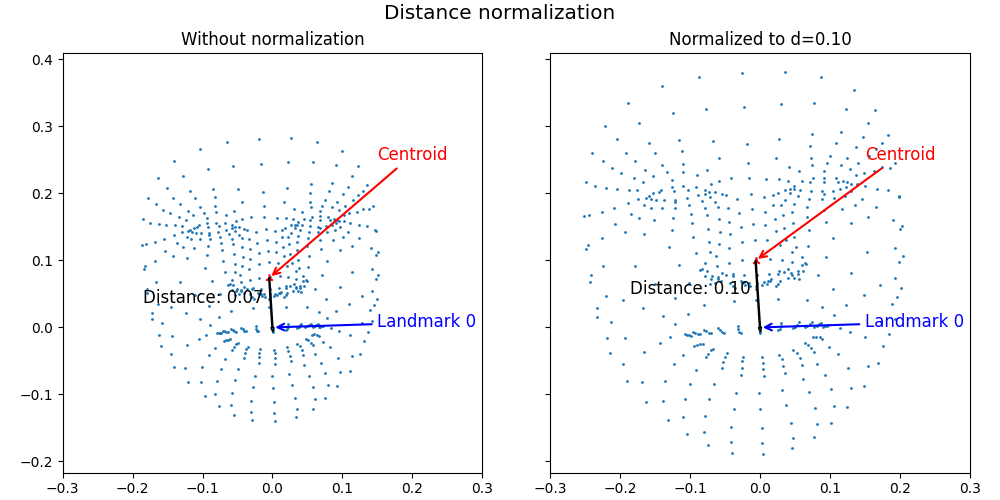


*Figure 3. Landmark extraction process. The image is extracted from the RAVDESS dataset.*

### Normalization of landmarks

While the landmarks are normalized with respect to the image size, they still include biases that can interfere in the classification such as the position of the face in the image or the distance from the camera. By doing this type of normalizations, students can observe the impact of these techniques in the result of the system, learning interactively its importance. As a baseline, a normalization for these two cases is proposed:

* The position of the face is normalized by displacing the landmark number 0 to the origin of coordinates. In order to achieve this, the coordinates of that landmark are subtracted from every other landmark. By doing this, we eliminate the bias of the position of the face in the image, fixing its position to the origin of coordinates.
* The distance of the user from the camera can also be a source of noise as the size of the face shrinks when the user moves away from the camera. To counter this, a scale of the landmarks is proposed. Specifically, the centroid of the landmarks is obtained and, from here, the landmarks are all scale by a factor *K* to obtain a fixed distance between the new centroid and the landmark 0. This transformation can be seen in Fig. 4.

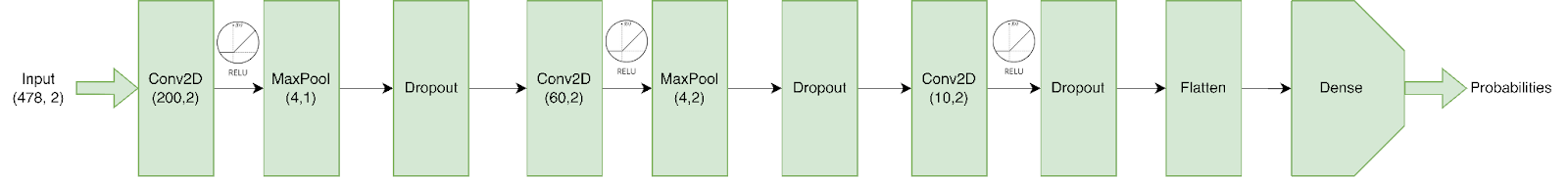


*Figure 4. Normalization example of the landmarks.*

# Model training and comparison

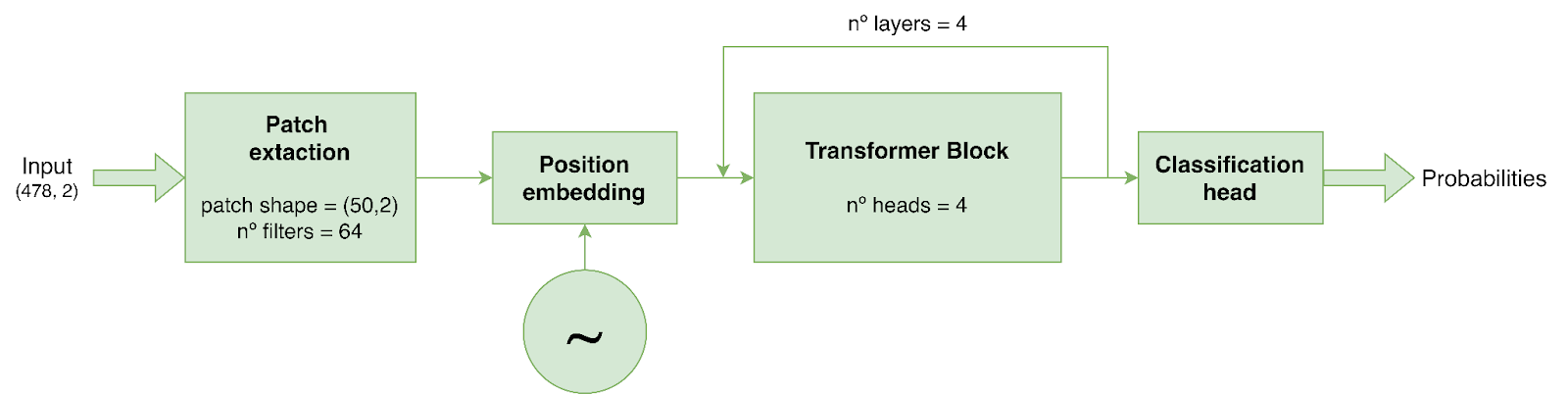
Once the data is loaded and preprocessed the training starts. In this phase, we pass the data to a model previously defined. During this process, the model will find characteristic patterns in the landmarks, which will be used to predict the emotion. Two different models are proposed as initial options: one Convolutional Neural Network (CNN) and one Visual Transformer (ViT).

The CNN proposed as baseline consists of three convolutional layers with max pooling layers between them. Dropout layers are inserted along the pipeline to help the model generalize better. The architecture of the model can be seen in Fig. 5.



*Figure 5. Architecture of the CNN model.*

The second model relies on the transformer architecture. These models apply attention mechanisms to detect the most important part of the input and, focusing on it, predict the correct class. Specifically, the ViT model first extracts patches from the images, to then pass them to the transformer as tokens. Finally, a classifier head is appended to the model. These architectures are becoming the state of the art in numerous tasks, making them a good option when searching for better results. They can also be configured with different hyperparameters, the most important ones are the patch size, number of filters, number of layers, and number of heads. As ViT base is also presented as the starting point in Fig. 6.



*Figure 6. Architecture of the ViT model.*

These two models can be trained both from scratch, just training them from the base model with one of their own recorded dataset or the RAVDESS dataset. Nevertheless, the students can also fine tune these models. From having a previously trained version with the RAVDESS dataset, they can tune the learnings to their own dataset, experiencing the benefits of both scenarios: a big dataset with much data and a smaller but more specific to their desired functionality. In both cases the models were trained using Adam optimizer and 'categorical\_crossentropy' as loss function.

Once they are trained, the models obtained can be compared. The comparison is made with the “accuracy” metric and takes into account a 95% confidence interval. Students can experience comparing their different approaches and selecting the best solution found for their application. In this line, training can be done by splitting the dataset into train and test, or by doing K-fold training. The latter allows for more consistent results as the confidence interval shrinks.

# RESULTS

The text included in the sections or subsections must begin one line after the section or subsection title. Do not use hard tabs and limit the use of hard returns to one return at the end of a paragraph.

## Subsection [Arial 12, bold, left alignment and capitalize the first letter]

Please, do not number manually the sections and subsections; the template will do it automatically.

### Sub-subsection: Guidelines for Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Do not use abbreviations in the title or heads unless they are unavoidable.

### Sub-subsection: Guidelines for Figures and Tables

Tables and figures should be centred and are numbered independently, in the sequence in which you refer to them in the text. Use the abbreviation “Fig. 1”, even at the beginning of a sentence. Figure captions should be below figures and graphics should be accompanied by a legend; table heads should appear above tables.

Table 1. Caption for the table.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Heading 1 | Heading 2 | Heading 3 |
| One | 1 | 2 | 3 |
| Two | 4 | 5 | 6 |
| Three | 7 | 8 | 9 |



Figure 1. Caption for the figure.

### Sub-subsection: Guidelines for Page numbers and Footnotes

Please, do not add any kind of pagination anywhere in the paper. Avoid using headers and footnotes.

### Sub-subsection: Guidelines for References

The list of the references should be given at the end of the paper. References are numbered in brackets by order of appearance in the document (e.g. [1], [2], [3]). The same reference can be cited more than once in the text with the same reference number. The references should be cited according to the Bibliography and Citation Style: <https://iated.org/citation_guide>.

# CONCLUSIONS

Use as many sections/subsections as you need.

ACKNOWLEDGEMENTS [Arial, 12-point, bold, left alignment]

Optional statement to thank other contributors, assistance, or financial support.

REFERENCES [Arial, 12-point, bold, left alignment]

References [Arial, 10-point, left alignment, upper and lower case] should be cited according to the Bibliography and Citation Style <https://iated.org/citation_guide>

1. A. Einstein, “General theory of relativity,” *Annalen der Physik*, vol. 49, no. 7, pp. 769–822, 1916.
2. A.A. Author, "Journal/Conference Article Title," *Periodical Title*, vol. Volume, no. Issue, pp.-pp., Publication Year.
3. A.A. Author, *Book Title*. City/State: Publisher, Year of Publication.
4. A.A. Author, "Chapter Title" in *Book Title* (Editors eds.), pp.-pp., City/State: Publisher, Year of Publication.
5. A.A. Author, "Online Article Title," *Periodical* *Title*, vol. Volume, no. Issue, pp.-pp., Publication Year. Retrieved from URL.