Pi² Closure report 2023/2024

Deep Learning Model and Knowledge Graph for Re-Identification of graphic assets in videos

This project focuses on the reidentification of graphical assets, known as ReId. ReId, commonly applied to individuals, is a computer vision task aiming to match the identity of a person across multiple instances extracted from video or image sequences. While existing deep learning techniques are effective with high-quality videos and easily identifiable individuals, this project addresses the challenge of detecting and tracking individuals across different images captured by one or more cameras using various features such as appearance, body shape or position, clothing. The objective is to compare traditional deep learning methods with an approach combining computer vision techniques and graph theory (Knowledge Graph, Graph Embedding, Graph Mining). The team of students will engage in all stages of a scientific research project, from conducting a literature review to implementing experiments and validating results, under the supervision of a doctoral candidate and ESILV professors.

Team Number: 283

Project Partner: Pierre Lefebvre, Nicolas Travers et Ahmed Azough

Project Team: Modèle de Deep Learning et Knowledge Graph pour la RéIdentification d'assets graphiques dans des vidéos.

Team Member:

- CORNELIE Joalie
- GIRAUD Louis-Melchior
- MARTYR Louis
- MICHON Raphaël
- MEZIANI Emrys
- LEROY Malo

Table of Contents

1 Objectives Set versus Results Achieved

1.1 Project Initial targets

The proposed project focuses on the re-identification of graphic assets, known as ReId. ReId, for people for example, is a computer vision task whose objective is to match the identity of a person to multiple instances taken from a video or image sequence. Deep Learning techniques exist but work on high quality videos and when people are easily identifiable.

The challenge of this project is to be able to detect and track a person, using characteristics such as appearance (in the form of descriptors called "feature vectors"), body shape or position, clothes, age or ethnic group to match their identity in different images captured by one or more cameras.

The objective is therefore to compare classic Deep Learning methods to a method combining computer vision and graph theory techniques (Knowledge Graph, Graph Embedding, Graph Mining). Students who choose this project will have the opportunity to participate in all the stages of a scientific research project from conducting a state of the art to implementing experiments and then validating the results under the supervision of a PhD student and professors from ESILV.

This project has been divided into 3 different milestones, composed as follows:

1. Literature review on the theme of person re-identification
    a. Inventory of the main re-identification models from the state of the art
    b. Inventory of the main datasets used
    c. Inventory of the main evaluation metrics for models
    d. Writing a summary of our research
2. Preparing the dataset for model training
    a. Choice and grouping of royalty-free videos from video surveillance systems
    b. Identification of the different formalisms and types of files for data labelling and choice of the latter for the following steps
    c. Data cleaning (detection and correction of incomplete, incorrect, or illegible data such as truncated files, blurry videos or content that does not match the initial theme)
    d. Data visualization to better understand the data we will be working with, and obtain preliminary notions of the various biases the model may potentially encounter
    e. Creating datasets structures for the model to use during its training
3. Model training
    a. Writing Python preprocessing scripts (importing the dataset, standardizing, splitting, data augmentation, etc.)
    b. Participation in designing a re-identification model and validation protocol then training the model
    c. Creation of a new model
    d. Validation of models and comparison of results with the literature

These missions slightly evolved throughout the year depending on the groups' progress, however the changes were very slim. Our project partner also added an extra step for the last milestone, which was to create a tracker that would automatically detect bounding boxes and use our previously constructed model's weights to apply the Re-ID on any video.

1.2 Results Achieved

I. Literature Review

We began our project by conducting an in-depth reading of a comprehensive scientific survey paper on Person Re-Identification (Re-ID). This survey served as a structured state-of-the-art review, compiled from multiple peer-reviewed scientific research documents.

The survey provided invaluable insights into the different existing technologies and techniques employed for re-identifying individuals across images or video footage.

Furthermore, it delved into the evaluation methodologies for re-identification models and presented an overview of the various publicly available datasets used for training and testing such models.

II. Dataset Analysis

Armed with the knowledge gained from the literature review, we proceeded to perform in-depth analyses on two widely used re-identification datasets: CAMPUS and PRW.

These datasets, weighing several gigabytes in size, comprised video surveillance footage captured from multiple cameras strategically placed in locations such as university campuses.

The PRW dataset contained over 10,000 images and featured more than 900 unique identities captured across 6 different cameras.

The CAMPUS dataset, on the other hand, consisted of video sequences recorded in 4 distinct locations, each approximately 4 minutes in duration.

Our analysis encompassed the import, comprehension, and transformation of the raw dataset files into a format suitable for further processing.

Additionally, we conducted a thorough exploratory data analysis through visualizations, enabling us to identify and address any potential issues, such as incorrect or unevenly distributed data points.

III. Dataset Preparation

Following the comprehensive analysis, we constructed the necessary dataset structures using the PyTorch deep learning framework, preparing them for the subsequent training of our re-identification model.

This step involved careful preprocessing of the data, including techniques such as data augmentation and normalization, to ensure optimal performance during model training.

IV. Model Training

Initially, our approach was to develop the re-identification model using PyTorch. However, upon the recommendation of our project supervisor, we pivoted to the TensorFlow framework for enhanced adaptability and compatibility with future extensions.

The model training process presented some challenges due to the limited RAM resources available on the Google Colab environment we were using.

Through meticulous optimization of our code and implementation of memory-efficient techniques, we successfully overcame these constraints.

As a result, we obtained a highly performant re-identification model, achieving an impressive accuracy of 0.92 on our test datasets.

V. Tracker Application

Initially, our focus was solely on developing deep learning models for person re-identification. However, our project supervisor found the idea of integrating our model into a person tracking system to be an interesting and valuable extension.

With a well-trained and accurate re-identification model at our disposal, we extracted the learned weights and parameters for future integration into a person tracking system.

This system will leverage the trained model to seamlessly track and re-identify individuals across multiple camera feeds or image sequences, enabling a wide range of applications in surveillance, security, and computer vision domains.

1.3 List of Deliverables

List and the content of our reports:

- A Google colab notebook for the steps 2 and 3 of the project;
- 3 PowerPoint presentations for the 3 CTS meetings we conducted throughout the year working on this project;
- The 3 corresponding reports for those 3 CTS meetings.

The documents listed above are all present in the first link added below, a Google drive containing all of them. They were stocked during the time we worked on the project on our own Microsoft Teams and they were seen, sent and corrected by our partner which is also our CTS, Pierre Lefebvre.

Here are the links to our deliverables and to the documents handed to us for the different steps of the project (second link):

https://drive.google.com/drive/folders/1Y0bhsugOJFKDiXihcRiMFYqfo9bN1qXl?usp=share_link https://drive.google.com/drive/folders/1XDL6F9LUXPFFWl0lQtjnsQ2Xnk7C2NNb?usp=share_link

2 Methodological reviews

During the beginning of our project, we had a Soft-skills meeting. Indeed, this meeting allowed us to draw up the MBTI of the team, which is I S T/F P. We had the opportunity to clarify during and after the positive points of our work, but also to identify areas for improvement and possible solutions. For example, to set intermediate deadlines other than those set by the partner so that we don't get our work done at the last minute. That's why we started to make a Gantt chart to have a visual schedule of our different milestones. Mainly due to our MBTI profile, the team's collaboration was simple and effective, we did not have team leaders but more people who offered to take initiatives to work or organize our work. Thus, the distribution of tasks was done firstly for milestone 1, but also for the preparation of the meetings according to the different points to be worked on and secondly for milestones 2 and 3 we set final objectives by imposing deadlines on ourselves. Everyone was free to work on a part according to their free time and technical knowledge to continue the work done before. This is why we can say that team communication was effective and allowed us to carry out a good project environment.

3 Risk Management

Legal: Image Rights

Risk: Our project involved the use of datasets that may contain copyrighted images, posing a risk of violating image rights.

Management Strategy: Verification of sources and rights of our datasets, and the creation of proprietary datasets.

Effectiveness: We didn't have the time nor the resources to create our own dataset, so we stuck with what our partner gave us.

Security: Data Breach

Risk: The potential for unauthorized access and disclosure of sensitive data.

Management Strategy: Implementation of robust data access protections.

Effectiveness: We didn't have that much sensitive data in the end since these datasets are all public, but we are conscious of the fact that this data could be used in nefarious ways if the camera sites are placed in some high priority locations.

Environmental: Model Training Consumption

Risk: The environmental impact due to the high energy consumption of model training.

Management Strategy: Utilizing processors with lower carbon footprints and favoring local machine processes over cloud-based solutions.

Effectiveness: We only used the Google colab free-to-use machine and were somewhat able to train the model, not without some complications though.

Technological: Technology Incompatibility

Risk: The incompatibility of deep learning model libraries with data retrieval software.

Management Strategy: Ensuring compatibility between all utilized technologies.

Effectiveness: The model was capable of handling the data we had in the end thanks to the help of our CTS who knew a lot about how to format it and write the model.

Operational: Dataset and Image Handling

Risk: Challenges related to the large size of datasets and images, including storage, processing, and memory constraints.

Management Strategy: Splitting datasets into smaller batches, reducing image sizes, and optimizing training subsets.

Effectiveness: This really helped to create and train the model, we had to really dive deep into each hyperparameter in order for the model to compile and not run for hours.

Hardware: RAM and Computational Time Limitations

Risk: Insufficient RAM and processing power for complex computations.

Management Strategy: Accessing premium accounts for better GPU usage or utilizing personal powerful computers.

Effectiveness: At the beginning, the model given by our CTS was eating all the RAM of the Google Colab machine, but we fine-tuned it and were able to overcome this problem.

4 Constraints

This project was mainly run using Google Collab. This brought a variety of constraints of multiple natures. First, a good internet connection was needed to access the project. Also, Google Collab does not support multiple people modifying the file at the same time, an issue that affected greatly the workflow. Solutions were to either work in parallel in 'local' and publish the changes from time to time, or to work at different times and relay ourselves.

As Collab is a web service, we also had to consider Google Terms of Service. It bans the sharing of a premium Collab account, as per the part 5 of the TOS (https://colab.research.google.com/pro/terms/v1). That means we couldn't use a premium account that would have given us more leeway concerning the following point, that was the most problematic constraint in our opinion.

In addition to software and legal limitations, Google Collab has a hardware limitation concerning the RAM attributed to a project. Our project needed to train a model on large, heavy datasets and the RAM we were allowed to use wasn't sufficient. The hardware given is also on a timer, with a session timeout after 12 hours of use meaning we couldn't keep the project running at night. This severely limited our group for this power-hungry project. Our personal hardware also contributed to this constraint, as no one in the team had a computer powerful enough to run the training on their hardware, leaving Collab the only option.

5 Technical Review

Most of the technical options we chose with regard to the functional specifications end up concerning the different libraries and methods we used to accomplish the tasks that were given to us. So here we will resume the most important ones:

- pandas : to store all of our data in easy-to-use datasets
- numpy / scipy : to do all sorts of mathematical applications to our data
- google.colab.drive / ZipFile : to import and extract the data instead of downloading it locally
- matplotlib / seaborn : to make all sorts of data visualizations
- cv2 / PIL.Image : to store and display images
- PyTorch / TensorFlow : to create our Re-ID model's and datasets' structure, train the model, evaluate it, etc…

Now let's dive in more specifically in our model training part. One of the most important technical challenge we had to face was concerning Google Colab's RAM limitations, that wouldn't allow us to load too much images at once for the training stage. Therefore we went through 3 different options to tackle this issue. We also had some issues with the GPU usage time provided by Colab, some times we had to change account or wait few hours to be able to use the GPU again.

1. Lazy loading

The first solution we thought of was to implement a Lazy loading method, where each image would be loaded one by one during the model's training and instantly removed when moving on to the next one. This solution ended up working correctly, however the training took a very long time to complete, and we couldn't tweak the hyperparameters much, as we would quickly again reach Colab's memory limits.

2. Load only bounding boxes

The second solution we tried was to load every single training images during the creation of our Datasets, but instead of loading the full images, we would only add the cropped bounding boxes to significantly reduce the space used and training time. However, this solution gave us very poor results for reasons we are still not sure about, so for now it is out of the question, but it might be interesting to take a deeper look at it.

3. Create a cropped copy of the dataset

The third solution was to create a copy of the dataset that contained the paths to the images cropped using the bounding box coordinates. All of these images were stored in another folder, following a naming convention based on the ID, camera, and frame. Furthermore, these images were lighter in size compared to the original ones. To enhance the data quality, we also performed cleaning on the unknown IDs. After implementing these steps, the custom model based on Resnet50 ran effectively around 25 epochs instead of 50 due to the early stopping, achieving an accuracy of around 0.92 on the test data, which is an excellent score.

Regarding resources for this project, no financial or technical resources were required. We all worked using the free version of Google Colab accessible through any Google account. Relevant research articles and example notebooks were also provided by our project partner, who was always available for questions.

Furthermore, some planned milestone deadlines were adjusted to better suit our progress and team members' schedules. However, the final project deadline remained as originally planned. While part of the final milestone went incomplete mainly due to the end of semester time constraints, we believe we could have finished it only shortly after the intended completion date.

6 Engineering standards

Throughout the project, adherence to various engineering standards and best practices played a crucial role in ensuring the effectiveness, efficiency, and ethical integrity of our work.

Data Handling and Privacy Standards: Throughout the project, we ensured compliance with data handling and privacy standards, particularly regarding the usage of sensitive data and potential legal ramifications. We adhered to established guidelines such as the GDPR (General Data Protection Regulation) in the European Union, and similar data protection laws where applicable, to maintain the integrity and privacy of the datasets used, including those sourced from PRW and CAMPUS.

Machine Learning Framework Standards: Our project extensively relied on industry-standard machine learning frameworks such as *PyTorch* and *TensorFlow*. These frameworks are renowned for adhering to best practices and standards within the machine learning community, ensuring robust model development, evaluation, and deployment processes.

Cloud Computing Standards: Utilizing Google Colab as our primary development platform necessitated adherence to cloud computing standards and best practices. This included considerations for security, data handling, and resource management within the cloud environment, ensuring the integrity and availability of our project resources.

Image Processing Standards: Given the involvement of image handling and processing tasks in our project, we adhered to established standards and best practices in image processing. These may include guidelines outlined by organizations such as ensuring accuracy, efficiency, and ethical handling of image data.

Project Management Standards: Although not strict engineering standards, our project adopted project management methodologies and best practices to effectively manage tasks, milestones, and resources. This likely involved employing a planning method such as a Gantt chart to ensure efficient project execution and alignment with project goals and deadlines.

7 "Post-project" tasks

The main task that is left for our project to be considered as complete, would be to implement the tracker and apply it using our best model's weights. Then we would have successfully created a fully functional Re-ID model.

8 Project Completion Recommendations

The key points identified in our closure report for the key recommendations are:

Best Practices:

- Thorough literature review to understand the problem.
- Detailed dataset analysis to ensure data quality.
- Optimizing code for efficient model training.

Areas for Improvement:

- Create proprietary datasets or verify image rights to avoid legal issues.
- Use hardware with lower environmental impact for model training.
- Ensure compatibility between software and libraries used.

Corrective Actions:

- Implement the person tracking system using the trained model.
- Continue optimizing dataset handling for memory constraints.
- Access more powerful hardware resources if needed.

The report also highlights our effective team collaboration, communication strategies (like using a Gantt chart and interim deadlines) and leveraging team members' strengths based on their personality profiles (MBTI). These can be considered best practices for future projects.