Machine Learning and Big Data Processing

Projects 2018

Adrian Munteanu, Nikolaos Deligiannis (course titulars)

Quentin Bolsee, Duc Nguyen (Tas)

Guidelines

The aim of your projects is to study, implement, test and demonstrate machine learning algorithms with applications in multimedia processing. You should program in Python using Tensorflow or SKlearn libraries, as taught during lab sessions.

- You have **to write your own Python code**. Additional libraries can be used for easier pre-processing of the data, but may not constitute the bulk of your project.
- **Directly copying existing code from external sources without acknowledgement disqualifies the project, leading to a project score of 0 for the entire group.**
- You have to submit a written project report of 6-10 pages.
- Each group submits ONE report.
- Sections (obligatory): Introduction, State of the Art, Description of your algorithm, Experimental results, Conclusions, References

- **Project submission**

  Submit as pdf (along with the code and a README.txt file describing how to run the code, all in a zip file) to:
  qbolsee@etrovub.be and mmdnguyen@etrovub.be

- **Grading**

  The project accounts for 40% of the final score for the image processing exam. The project grading is based on:

  - Technical quality (score per group)- 40%
  - Written report (score per group)- 25%
  - Oral defense (individual score)- 35%

  The oral presentation and Q&A (~20 minutes) will be done in groups with a single Power point presentation. Your score will be based on your understanding of the image processing techniques you used in your project.

  The deadline and date for the oral defense will be announced soon.

1. Image super resolution

Description

Super resolution from a single low resolution image has always be an ill posed problem. The current state of the art consists in training a convolutional neural network with a large dataset of natural looking images. Corresponding low resolution images are computed and fed as an input to the neural network. The learned feature maps have shown to produce less artifacts than classical methods.

Objectives

The first step in this project is to retrieve the dataset. We suggest using the DIVerse 2K resolution high quality images[1] which provides both high and low resolution images.

The most interesting aspect of this project is the method to upscale the number of pixels in a given layer of your network. A very powerful approach is to interleave pixels of separate feature maps in a so-called sub-pixel convolution. Other works work on a bicubic interpolation of the low-resolution image to avoid this, or use sparse-coding to map low-resolution patches to high-resolution ones.

After training, compute the PSNR of the reconstructed image on a validation dataset.

The deliverables of this project are : (i) a brief description of the literature ; (ii) a working super resolution CNN with a few different architecture parameters; (iii) an experimental study of the reconstructed image PSNR against plain bicubic interpolation.

References

- Shi, Wenzhe, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1874-1883. 2016.
- Dong, Chao, Chen Change Loy, Kaiming He, and Xiaoou Tang. "Learning a deep convolutional network for image super-resolution." In *European Conference on Computer Vision*, pp. 184-199. Springer, Cham, 2014.

[1] https://data.vision.ee.ethz.ch/cvl/DIV2K/

2. Music classification

Description

The recent advances in machine learning and especially deep learning enable several applications in multimedia. A very interesting application is automatic analysis of musical contents. Some examples are *Shazam* and *Spotify Discover* – online content-based music discovery services. These services belong to the category of music retrieval problem. A related problem, which has also received a lot of attention is *content-based Music Classification.* This problem involves identifying the characteristics of a song from the content (i.e. rhythm rather than metadata). Interesting characteristics include *genre, artists, era* and so on. These characteristics are useful for applications such as automatic organization of music collections, music recommendation and also retrieval.

Objectives

In this project, you can choose to infer one specific characteristic from musical signals. As a suggestion, you can work on the *genre classification* problem, as its literature is quite rich. Nevertheless, we encourage your to have your own proposal and creativity.

A basic pipeline to approach this problem contains two steps : feature extraction and classification models. As of the dataset, a suggestion is to use a subset of the Million Song dataset[2]. You can use an external library to process audio with Python[3].

Upon completing this project, we expect : (i) a brief description of the literature ; (ii) a working demo with different feature extraction methods and at least one classification algorithm ; (iii) an experimental study with different components you have built.

References

- v. d. Oord and S. Dieleman and B. Schrauwen, Deep content-based music recommendation", *Internal Conference on Neural Information Processing Systems*, 2013
- P. Ahrendt, "Music Genre Classification Systems - A Computational Approach", 2006
- T. Li and M. Ogihara and Q. Li, "A Comparative Study on Content-based Music Genre Classification", *Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, 2003, pp. 282-289.
- Other papers found on Google Scholar

[2] https://labrosa.ee.columbia.edu/millionsong/
[3] https://github.com/librosa/librosa

3. Image denoising

Description

When measuring real-world color values, any digital imaging sensor will be disturbed by noise. When correcting the value of a given pixel, methods involving the weighted average of neighboring pixels are often used. However, it has recently been shown that the mapping from a noisy signal to a noise-free version can be learned very efficiently with a convolutional neural network. The non-linearity of this mapping outperforms classical signal processing methods. The goal of this project is to train a CNN to the task of removing additive gaussian noise on grayscale images.

Objectives

For the image dataset, consider using the Berkeley Segmentation dataset[4]. You can add generated Gaussian noise of a fixed variance to produce the noisy images yourself.

An effective method is called *residual learning* : the output from the CNN is not the denoised image, but the noise pattern alone. It can then easily be subtracted from the noisy image to produce the final result. You should compare your results with and without residual learning.

Upon completing this project, we expect : (i) a brief description of the literature ; (ii) a working neural-network-based denoising method; (iii) experimental evaluations with different model architectures.

References

- Zhang, Kai, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising." *IEEE Transactions on Image Processing* 26, no. 7 (2017): 3142-3155.

- Burger, Harold C., Christian J. Schuler, and Stefan Harmeling. "Image denoising: Can plain neural networks compete with BM3D?." In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 2392-2399. IEEE, 2012.

[4] https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/

4. OCR for handwriting

Description

Handwritten character recognition has been one of the most basic and popular problems in Computer Vision for a long time. The task is to infer the correct label (letter or digit) given an image of handwritten character. This problem, often called *Optical Character Recognition (OCR)*, has been employed in a lot of application. Nowadays, we can see OCR system in our smartphones (Android and iOS), or even in our PDF reader.

Objectives

In this project, you will build an OCR system, employing Convolutional neural network models. You are recommended to use the EMNIST dataset[5] (the full dataset or a subset of it, depending on your computational resources).

Upon completing this project, we expect : (i) a brief description of the literature ; (ii) a working OCR system, which can recognize our handwritten letters, captured by camera (you will need to perform some pre-processing to make it similar to images you have in the EMNIST dataset) ; (iii) an experimental evaluation of the model with different configurations.

References

- G. Cohen, S. Afshar, J. Tapson and A. van Schaik, "EMNIST: Extending MNIST to handwritten letters", *International Joint Conference on Neural Networks (IJCNN), 2017, pp.2921-2926*

- Y. Lecun and L. Bottou and Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition", *Proceedings of the IEEE*, vol. 86, no. 11, 1998, pp. 2278-2324

[5] https://www.nist.gov/itl/iad/image-group/emnist-dataset

5. De-anonymization

Description

In the era of big data and the Internet, people are publishing more and more traces to their real personal information online, most of the time unintentionally. This often happens through mobile apps, public websites and especially social networks. To gain publicity, service providers often publish their users' data, such as reviews, opinions, sharings. To assure users of the privacy safety, these providers often perform the anonymization process before publishing data about their users. Nevertheless, as each service provider can only anonymize their data independently from other services, attackers can cross-correlate data from different sources to identify more personal information about the users or even their real identities. This is called the ***de-anonymization*** or ***re-identification*** attacks.

Objectives

In this project, in a team, you will need to study the literature and produce a proof-of-concept of a de-anonymization attack. As a suggestion, you can reproduce the Netflix-IMDB attack, for which the data is realatively easy to access. Nevertheless, you are encouraged to have your own choice of data and method to work on, from your literature study.

Upon completing this project, we expect : (i) a brief description of the literature ; (ii) a working proof-of-concept of an attack.

References

- Narayanan, Arvind & Shmatikov, Vitaly. (2006). "Robust De-anonymization of Large Datasets (How to Break Anonymity of the Netflix Prize Dataset)"
- Papers found on Google Scholar with related keywords, e.g. de-anonymization, re-identification, …

6. Voice isolation

Description

In human perception, a sound source is easily isolated, even in a crowded environment. A showcase task in this context is to isolate the singer's voice in a song. Neural Networks have been shown very effective for this task, thanks to their flexibility. Typically, we work on a spectrogram version of the sound, and require the network to produce a mask for each frequency bin. This mask indicates whether the bin belongs to the singer or the instruments.

Objectives

The first step in this project is to build a music dataset, along with the singer audio track isolated from the instruments. For this purpose, we suggest using the MedleyDB dataset[6] which contains individual tracks for 122 songs.

Short time Fourier transform must be used to produce a spectrogram of the sound, that can then be processed by a neural network. You can use an external library to process audio with Python[7].

Upon completing this project, we expect : (i) a brief description of the literature ; (ii) a working demo of a singer isolation

References

- Simpson, Andrew JR, Gerard Roma, and Mark D. Plumbley. "Deep karaoke: Extracting vocals from musical mixtures using a convolutional deep neural network." In *International Conference on Latent Variable Analysis and Signal Separation*, pp. 429-436. Springer, Cham, 2015.

- Other papers found on Google Scholar

[6] http://medleydb.weebly.com/
[7] https://github.com/librosa/librosa

7. Matrix completion

Description

Matrix completion is one of the key problems in machine learning, It concerns the problem of recovering a matrix from partial observations, where a matrix cannot be fully sampled or directly observed. Examples of machine learning tasks that employ matrix completion algorithms include clustering, classification, and especially recommender systems. The Matrix Factorization (Collaborative Filtering) exercise that you saw during the lab sessions is also a Matrix completion method. Recently, deep neural network based solutions have been proposed to tackle the matrix completion problems, and they are achieving state-of-the-art results nowadays.

Objectives

In this project, you will study the literature of neural-network-based matrix completion and implement and experiment with one model. Concerning the dataset, you can select and use (at least two) from : different versions of the MovieLens dataset[8], the Book-Crossing dataset[9], the Jester dataset[10] and the Netflix dataset[11].

Upon completing this project, we expect : (i) a brief description of the literature ; (ii) a working neural-network-based matrix completion model ; (iii) experimental evaluations with different model architectures.

References

- J. Fan and T. Chow, "Deep learning based matrix completion", Neurocomputing (266), pp. 540-549.
- S. Suvash & A. K. Menon and S. Scott and X. Lexing, "AutoRec: Autoencoders Meet Collaborative Filtering", WWW 2015, pp.111-112
- F. Strub and R. Gaudel and J. Mary, "Hybrid Recommender System Based on Autoencoders", Workshop on Deep Learning for Recommender Systems, 2016, pp.11-16.

[8] https://grouplens.org/datasets/movielens/
[9] https://grouplens.org/datasets/book-crossing/
[10] https://grouplens.org/datasets/jester/
[11] https://archive.org/download/nf_prize_dataset.tar

8. Grayscale image colorization

Description

A large amount of historical grayscale images is available online. Adding color information to an existing image can be done by an artist, but this is time consuming. The most effective methods can now assist the artist when propagating colors on the whole image. Colorization with a neural network alone has also been explored, and showed interesting results. The goal of this project is to build your own colorization CNN.

Objectives

In this project, you should start by building a grayscale/color image dataset from a public natural image dataset such as Imagenet[12] or the Berkeley Segmentation dataset[13].

The learned CNN produces a chrominance image (UV space) from a luminance image (Y only). A typical issue that occurs when training the CNN is that most datasets are poorly balanced in the chrominance space: low values are more often present. An ingenious way to tackle this is to discretize the chrominance space, and formulate the problem as per-pixel classification task. Low occurrence of some classes (colors) in the dataset can be compensated by weighting the loss-function for those classes.

Upon completing this project, we expect : (i) a brief description of the literature ; (ii) a working neural-network-based colorization model; (iii) experimental evaluations with different model architectures.

References

- Zhang, Richard, Phillip Isola, and Alexei A. Efros. "Colorful image colorization." In *European Conference on Computer Vision*, pp. 649-666. Springer, Cham, 2016.
- Cheng, Zezhou, Qingxiong Yang, and Bin Sheng. "Deep colorization." In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 415-423. 2015..

[12] http://image-net.org/
[13] https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/