Applied machine learning system ELEC0134 22/23 report SN: 17062315

1 Abstract

Artificial Intelligence is evolving faster than ever. One domain in particular, computer vision has benefited from a vast amount of data annotated in the past decades and many recent breakthroughs on the software level.

It is now time to develop our own tool built on these discoveries. Its purpose is to tackle challenges in the face recognition area by analysing multiple features of the human face such as its shape, eyes and mouth which are essential to identify individuals and their emotions.

To reach this objective, we rely on an important source of annotated data in the form of pictures of celebrities and cartoonish characters.

The information from these datasets are then extracted and cleaned before being trained by various algorithms such as Support Vector Machine and Neural Networks which imitate the way human learn in order to make the algorithm able to differentiate individuals.

Results are obtained by testing the trained algorithm against a subset of the dataset left aside for this specific purpose.

Upon completion of this project, it should be possible to take these proofs of concepts and embed them in more complex systems such as smart cameras and robots which are the most common kind of devices which may benefit from an improved perception.

2 Introduction

Last century, science fiction authors such as Isaac Asimov predicted the rise of intelligent devices [1] [2] and now, the dawn of artificial intelligence is upon us. After, the Tay bot [3], it is now the turn of ChatGPT to earn an immense popularity among tech enthusiasts with its ability to provide detailed answers to any question [4] [5]. And to add to this triumph, Microsoft has shown interest in acquiring OpenAI, the creator of ChatGPT and many other AIs [6].

But while everyone has their eyes on this new chatbot, a more discrete, albeit, just as interesting revolution, has been ongoing for a few decades already in the computer vision domain.

The first visible step toward it was the introduction of captcha [7] in 1997. These challenge-response tests quickly became annotation tools channelling the efforts of humans to train artificial intelligences [8].

Applications are numerous, ranging from biometric tests to disease detection [9] and even robotics with the latest self driving cars [10].

In this report, we shall pay close attention to a specific use case: face recognition. After mentioning the latests relevant breakthroughs, we shall see how we can design softwares able to recognise the gender, emotions, face shape and eye colour of various celebrities and cartoonish characters.

3 Literature survey

Face recognition has been a major topic recently with new algorithms and tools appearing in quick successions as early as last century with, for instance, the creation of Support Vector Machines in 1992, algorithms shining for their adaptability and generalisation ability [11] [12] or OpenCV created in 2000 and which is a library still widely used in many languages such as C++ and Python [13].

Many algorithms were used over the years such as Support Vector Machine and K-nearest neighbour [14]. They all come with their advantages and trade-offs. Convolutional neural networks especially tend to be studied with great attention because of their ability to imitate the human mind which has been AI's goal since its inception [15] [16].

Since various algorithms exist, the main objective nowadays shifted from pure scientific research to engineered solutions to deal with edge cases where computer vision still struggles [17].

Firstly, faces become harder to analyse with low resolution images and and dim lighting which make data extraction less precise [18].

Moreover noise is more important with data obtained from pictures of people belonging to specific demographic subgroups such as old males, people with darker skin types [19] or children which tend to appear as androgynous since gender only becomes more apparent past puberty [20].

These problems can generally be also explained by biased datasets which contain unbalanced representation of said demographic subgroups [21].

It is therefore essential to handle data preprocessing with as much care as the analysis itself as we will see in the next sections.

4 Models Description (chart, fig, eq explain/justify)

4.1 Support Vector Machines

With a linear SVM, we seek w and b such that $\rho = \frac{2}{\|w\|}$ is maximised and for all $(x_i, y_i), i = 1...n : y_i(w^T x_i + b) \geq 1$.

The quadratic operation problem can therefore be reformulated as finding w and b such that: $\phi(w) = w^T w$ is minimised for the aforementioned values of x and y.

With that in mind, we just need to solve a dual problem with a Lagrange multiplier α_i with
$w = \sum \alpha_i y_i x_i$ and $b = y_k - \sum \alpha_i y_i x_i^T x_k$ for any $a_k > 0$. It can be summarised as $f(x) = \sum \alpha_i y_i x_i^T x + b$ with every non-zero a_i indicating the presence of a support vector in the form of x_i.

4.2 Bagging

4.3 KNN

4.4 Random Forest

5 Implementation (Details. Explain key modules in code)

5.1 Task A1

5.2 Task A2

5.3 Task B1

5.4 Task B2

6 Experimental Results and Analysis

6.1 Task A1

6.2 Task A2

6.3 Task B1

6.4 Task B2

7 Improvements (analysis)

The aforementioned algorithms used for the binary and multiclasses tasks have varying degrees of success.

Preprocessing was crucial to avoid noise and underfitting with some algorithms which would have lowered their success below the acceptable threshold.

7.1 Support Vector Machines

Regarding Support Vector Machines, the algorithm performed relatively well with task A but could have been improved with a few tricks.

- A better processing for instance entailing standardising inputs properly would be beneficial.

- The same is true about trying different kernels such as the RBF one which perform very well in many situations or even kernels tailored specifically for the task at hand.

- Additionally, fine tuning with weight alteration and modification of the cost function always remains an option as long as it does not lead to overfitting by making the result mostly correlated to a subset of the training data

7.2 Convolutional Neural Network

In order to get better results with a CNN, it may be useful to try out the following ideas:

- Try out pre-trained models from the EfficientNet-B which are declined in 8 different versions (from 0 to 7) with incrementally more parameters and a common interface making them easily swappable for quick testing sessions.

- K-fold validation may be particularly useful with various models such as the ones mentioned previously. This strategy relies on splitting the dataset in K groups before selecting all but one for training and using for every iteration a different one for validation, hence getting more consistent results. Note that this represents an example of ensemble learning, a strategy consisting in mixing multiple domains in various ways.

8 Conclusion

Hence Support Vector Machines and Convolutional Neural Networks are algorithms revered for their high efficiency which was highlighted in those binary and multiclasses tasks related to face recognition.

It was essential to pre-process the data with ...

SVMs operate ... and really shine ...

On the other hand, CNN ... but are just as efficient ...

We may try next time to mix strategies with different strategies such as K-means clustering which is extremely slow and therefore inefficient but which has the advantage of converging with absolute certitude.

Moreover, a more promising alternative would be to try more Deep Learning strategies which imitate the way human learn and work particularly well in tandem with neural networks.

Thankfully, the results gathered from the present research should pave the way to better softwares combining them with other strategies such as reinforcement learning in robots and self driving car which could therefore both recognise humans and learn how to drive around them.

Figure 1: Caption

```
# test
if True == 1 + 1:
    pass

def test

class x
```

References

[1] Isaac Asimov and Rowena Akinyemi. *I, robot*. University Press, 2000.

[2] Ana Echeberria. “The Impact of AI on Business, Economics and Innovation”. In: Jan. 2022, pp. 67–96. ISBN: 978-3-030-88240-2. DOI: 10.1007/978-3-030-88241-9_3.

[3] Gina Neff and Peter Nagy. *Talking to Bots: Symbiotic Agency and the Case of Tay*. Tech. rep. Oct. 2016, pp. 4915–4931.

[4] Rexhep Shijaku and Ercan Canhasi. *ChatGPT Generated Text Detection*. Tech. rep. Jan. 2023. DOI: 10.13140/RG.2.2.21317.52960.

[5] Emmanuel Opara. *Chatgpt for teaching, learning and research: prospects and challenges*. Tech. rep. Jan. 2023. DOI: 10.13140/RG.2.2.21304.42244/1.

[6] Derek Saul. “Microsoft Reportedly Closing In On $10 Billion Investment Into ChatGPT Creator OpenAI”. In: *Forbes* (Jan. 2023). URL: https://www.forbes.com/sites/dereksaul/2023/01/10/microsoft-reportedly-closing-in-on-10-billion-investment-into-chatgpt-creator-openai/.

[7] Eran Reshef, Gil Raanan, and Eilon Solan. *Method and system for discriminating a human action from a computerized action*. Tech. rep. May 2005.

[8] Gaurav Goswami et al. “Face recognition CAPTCHA”. In: Sept. 2012, pp. 412–417. ISBN: 978-1-4673-1384-1. DOI: 10.1109/BTAS.2012.6374608.

[9] Zuber Khan, Tanay Shubham, and Ravi Kumar Arya. “Skin Cancer Detection Using Computer Vision”. In: Apr. 2022, pp. 3–11. DOI: 10.1007/978-981-19-0745-6_1.

[10] Emil Talpes et al. “Compute Solution for Tesla’s Full Self Driving Computer”. In: *IEEE Micro* PP (Feb. 2020), pp. 1–1. DOI: 10.1109/MM.2020.2975764.

[11] Mustafa Al-Dabagh, Salar Rashid, and Muhammad Ahmad. "Face Recognition System Based on Wavelet Transform, Histograms of Oriented Gradients and Support Vector Machine". In: 10 (July 2020).

[12] Zhao Jun. "The Development and Application of Support Vector Machine". In: *Journal of Physics: Conference Series* 1748 (Jan. 2021), p. 052006. DOI: 10.1088/1742-6596/1748/5/052006.

[13] J. Manikandan et al. "Face Detection and Recognition using Open CV Based on Fisher Faces Algorithm". In: *International Journal of Recent Technology and Engineering (IJRTE)* 8 (Jan. 2020), pp. 1204–1208. DOI: 10.35940/ijrte.E5735.018520.

[14] Sanmoy Paul and Sameer Kumar Acharya. "A Comparative Study on Facial Recognition Algorithms". In: *First Pan IIT International Management Conference* (Dec. 2020). URL: http://dx.doi.org/10.2139/ssrn.3753064.

[15] Abhishek Pandey et al. "Face Detection Using Convolutional Neural Network". In: Jan. 2023, pp. 673–679. ISBN: 978-981-19-1905-3. DOI: 10.1007/978-981-19-1906-0_55.

[16] Tijana Vukovic et al. "Thermal Image Degradation Influence on R-CNN Face Detection Performance". In: Nov. 2019, pp. 1–4. DOI: 10.1109/TELFOR48224.2019.8971128.

[17] Pawan Kumar. "Face Reading with AI (Artificial Intelligence)". In: (Dec. 2022).

[18] Mozhdeh Rouhsedaghat et al. "Face-Hop: A Light-Weight Low-Resolution Face Gender Classification Method". In: Feb. 2021, pp. 169–183. ISBN: 978-3-030-68792-2. DOI: 10.1007/978-3-030-68793-9_12.

[19] Samuel Dooley et al. *Robustness Disparities in Face Detection.* Tech. rep. Nov. 2022. DOI: 10.48550/arXiv.2211.15937.

[20] Sumithra Ram, Devanur Guru, and Manjunath Aradhya. "Face Image-Based Gender Classification of Children". In: Jan. 2023, pp. 213–228. ISBN: 978-3-031-22404-1. DOI: 10.1007/978-3-031-22405-8_17.

[21] Surbhi Mittal et al. *Are Face Detection Models Biased?* Tech. rep. Nov. 2022. DOI: 10.48550/arXiv.2211.03588.

9 Appendix

This is the output obtained when running the four tasks with 5000 images.

9.1 A1

```
: Working on task A1 with label gender from dataset celeba
: Proceeding to get 5000 images including 0.75% for training
:
: 100% 5000/5000 [01:57<00:00, 42.63it/s]
: Bagging (n=9) 0.5975
: KNN (n=5) 0.7
: Random Forest 0.8275
: SVM (poly) 0.9141666666666667
```

9.2 A2

```
: Working on task A2 with label smiling from dataset celeba
: Proceeding to get 5000 images including 0.75% for training
:
: 100% 5000/5000 [01:57<00:00, 42.66it/s]
: Bagging (n=9) 0.8241666666666667
: KNN (n=7) 0.8558333333333333
: Random Forest 0.8783333333333333
: SVM (poly) 0.8941666666666667
```

9.3 B1

```
: Working on task B1 with label face_shape from dataset cartoon_set
: Proceeding to get 5000 images including 0.75% for training
:
: 100% 5000/5000 [08:58<00:00,  9.29it/s]
: Bagging (n=1) 0.3209028459273798
: KNN (n=8) 0.4867517173699706
: Random Forest 0.6722276741903828
: SVM (poly) 0.7360157016683022
```

9.4 B2

```
: Working on task B2 with label eye_color from dataset cartoon_set
: Proceeding to get 5000 images including 0.75% for training
:
: 100% 5000/5000 [08:59<00:00,  9.27it/s]
: Bagging (n=9) 0.28361138370951916
: KNN (n=9) 0.28949950932286556
: Random Forest 0.34151128557409227
: SVM (poly) 0.3758586849852797
```