Recognising Image Shapes from Image Parts, not Neural Parts

Kieran Greer, Distributed Computing Systems, Belfast, UK.
kgreer@distributedcomputingsystems.co.uk
Version 1.0

Abstract - This paper describes an image processing method that makes use of image parts instead of neural parts. Neural networks excel at image or pattern recognition and they do this by constructing complex networks of weighted values that can cover the complexity of the pattern data. These features however are integrated holistically into the network, which means that they can be difficult to use in an individual sense. A different method might scan individual images and use a more local method to try to recognise the features in it. This paper suggests such a method, where a trick during the scan process can not only recognise separate image parts, as features, but it can also produce an overlap between the parts. It is therefore able to produce image parts with real meaning and also place them into a positional context. Tests show that it can be very accurate, scoring 100% on some handwritten digit test datasets. It also fits well with an earlier cognitive model, and an ensemble-hierarchy structure in particular.

Keywords: image classifier, image part, quick learning, feature overlap, positional context.

1 Introduction

This paper describes an image processing method that makes use of image parts instead of neural parts. It is conjectured that this method is more ‘intelligent’ than a traditional neural network, where the image parts that it creates not only have more meaning, but they can also be put into a positional context and allow for an explainable result. Neural networks excel at image or pattern recognition and they do this by constructing networks of weighted values that can cover the complexity of the pattern data. These networks recognise similarities in the data and resolve that into features which are shared between the patterns. These features however are integrated holistically into the network, which means that changing a feature can have unexpected consequences and they can be difficult to use in an individual sense. A different method might scan individual images and use a more local method to try to recognise the features in it. This paper suggests such a method, where a trick during the scan process can not only recognise separate image parts, as features, but it can also produce an overlap between the parts. This is very helpful and it means that the image parts can be placed into a positional context with each other. Then when comparing with a new image, it can be similarly parsed, when the image parts need to be in the same relative position, to be compared with each other. The process is intended to recognise image shapes, more than internal colours or textures, but this is still a difficult and challenging task. The tests of section 4 have been carried out on handwritten digit characters, and as noted in [6], handwritten character classification is fundamental in postal sorting, bank check recognition, automatic letter recognition, industrial automation, human-computer interaction, and historical archive documents. Initial tests suggest that this new method is highly accurate and it fits well with an earlier cognitive model, and an ensemble-hierarchy structure in particular.

The rest of this paper is organised as follows: section 2 gives some related work, while section 3 describes the new classifier in more detail. Section 4 describes some implementation details and test results, while section 5 gives some conclusions to the work.

2 Related Work

Deep Learning [12][13][15] has managed to almost master image recognition, but Decision Trees [4] are not far behind. At the heart of Deep Learning and the original Cognitron, or Neocognitron architectures [5], is the idea of learning an image in discrete parts. Each smaller part is an easier task and cells can then be pooled into more complex cells with neighourhoods. The deep learning architecture of [12] ends up with a top two layers that form an undirected associative memory, for example. Another image-processing algorithm was tried in [3] to recognise the letters dataset used later in section 4. A convolution can exaggerate a feature through a local transformation, or convert an image into one that represents the feature more. The paper [6] describes some other shallow architectures that include convolutions. They suggest a new Fukunaga–Koontz network that would process images more orthogonally and locally, but with the more advanced neural network architecture. However, the paper does recognise the goals of this paper when producing their new network structure.

The image classifier has derived from earlier work by the author, including the papers [8][9][10][11]. The paper [11] gives a first version for the algorithm, using only cell relations. Treating each pixel as a cell requires it to have a weighted association with the other pixels, which in that paper spaned the whole image. For example, a grid cell would map all of the other cells in an image it was present with, as a type of cross-referencing, to represent the cell importance with the desired category. There is no overlap with single cells, but using larger areas gives the region some definition that can make it both distinct and allow for the overlap. Using local regions therefore, is more similar to convolutions, where a second paper [10] showed that it can produce a reasonably useful autoassociative classifier, but that it does not generalise very well with previously unseen data. It did show however that a local calculation can replace the fully-connected weight values. That idea has then been used in this paper to produce the image parts, as described in section 3.1. The papers [9][11] introduced the idea of an ensemble-hierarchy structure, while the paper [8] used it as part of a self-organising system. The ensemble-hierarchy structure and also the self-organising unit can be realised as part of the image-processing structure of this paper.

There is some evidence that the scanning process of this method may mimic human eye movements. There are different types of eye movement [2], including smooth tracking movements or saccadic irregular movements, to fixate on and recognise features. These more irregular movements are what the new algorithm would make use of. It is interesting that the paper also writes about neural binding, as part of feature integration, which has also been studied as part of the cognitive model. The paper [14] suggests using attentional models instead of deep neural networks. It states that the computational expense of neural networks scales with the dimensionality of input images and can become prohibitive. Attentional models recast computer vision as a sequential decision-making problem, allowing an agent to deploy a sensor (i.e., an attentional window) to image data across multiple time-steps and the approach bears a resemblance to perceptual psychology. The paper's results demonstrate that carefully chosen models of visual attention can increase not only the efficiency, but also the accuracy of scene classification. Another paper that might have biological relevance is [16], which suggests that the hierarchical organization of the human visual system is critical to its accuracy. This is a functional hierarchy rather than a tree shape, however. While neural networks can learn this, they require orders of magnitude more examples than a human, who can accurately learn new visual concepts from sparse data, sometimes just a single example. Inherent in this then is the idea of orthogonality, rather than a statistical process of extracting shared features from a large dataset. They do then however, use deep learning as part of the architecture, to build the hierarchy of prior knowledge and exemplars.

3 The Image Recognition Classifier

This section describes a new type of classifier for image recognition. It divides an image into distinct parts, by processing each image individually, not as part of a statistical set. Then to create the classifier and help to economise, the image descriptions can be clustered into exemplar sets for each category type, when an exhaustive search over these exemplars can correctly classify previously unseen images.

3.1 Image Parts

The new algorithm is based on the fact that scanning over an image will automatically separate its shape into discrete parts, but it depends on the angle at which the image is scanned. If the scan is done in a vertical or a horizontal direction, then the whole image is returned. If it is done at an angle, or a circular direction, then the image can be separated into parts. It might be along the lines of irregular eye movements, for example. Current tests show that these parts describe distinct features in the external image shape only, they would not be useful for recognising internal patterns, for example. But one aspect of them is that the parts can recognise regions where lines join with each other. This type of recognition might require real intelligence, to properly understand what the image is about, but the scanning trick is able to realise these secondary features for itself. Figure 1 is a nice example that gives the image parts generated for a number ‘4’.

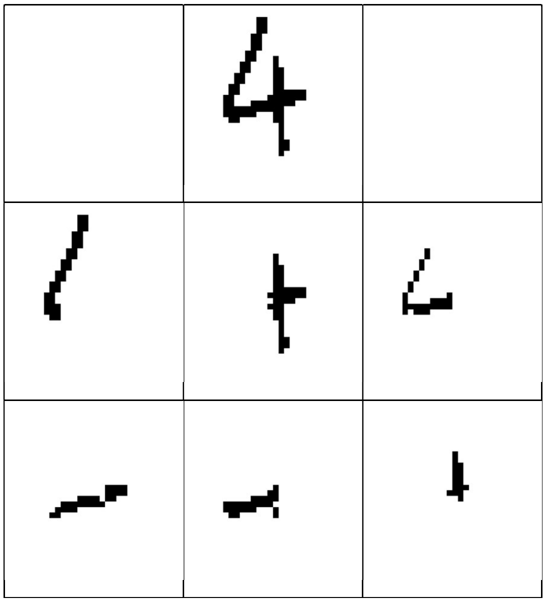

Figure 1. Image parts (rows 2 and 3) generated for a number 4 (row 1).

The scanning process makes use of the ideas of a continuous sequence and also convolutions [6][13], or producing an aggregated score from a region. The idea of cell associations was central to the first algorithm of [11] and subsequent work, and it is really only a count of what other cells are present when the cell in question is present. For recognising features, the scan counts the number of continuous cells before an empty cell is encountered, in the indicated direction. This helps to recognise the lines in the image, where the distances are then aggregated into cohesive sets. But at the moment, this is only for binary images with a 1 or 0 value in each cell. After a scan in a particular direction, cells with similar scores can be grouped together as an image part, and in fact, scans in different directions can also be cumulated together. In Figure 1, the image parts in rows 2 and 3 can be of different sizes, where they also overlap and the overlap can include joining regions, such as where the main horizontal and vertical lines join. This is close to all of the information that a human would require to understand the image shape.

3.2 Creating Exemplars

After this process, each image can be defined by a set of image parts. It would be possible to use the images like that, or it is possible to try to create exemplars from them. This may reduce the number of images to consider, when searching for a category. Clustering the images into exemplars can be a self-organising process [8] as follows: A distance can be measured for the closest images between categories. Then when combining images inside of a category, the distance between them must be less than the minimum distance to any image in any other category. This can reduce the total number of images by a small amount at least and produce a cluster or aggregated result of the image set, as an exemplar.

3.3 Ensemles-Hierarchy Structure

Each image for a category is now represented by a set of parts, either for itself, or for some aggregated result from a cluster. These parts could be placed into an unordered ensemble, for example, where they might even be reused as part of other images. That is a direction for future research on the cognitive model, but as a first step, it is possible to order the image parts further. The image parts can be ordered on size, which can then be used to create a hierarchical structure. The main body of the image is typically the largest part and then smaller distinct features around the edges of it are recognised. The cognitive model in earlier papers decided that the hierarchy part of the ensemble-hierarchy structure should in fact converge from the ensemble base to a single node furthest away, almost like a self-contained unit [7], shown again in Figure 2.

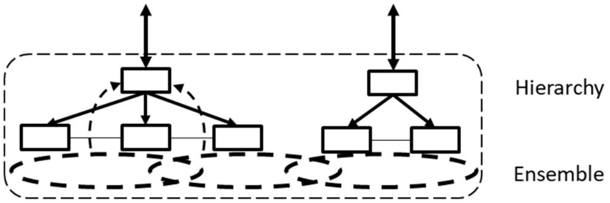

Figure 2. Overlapping ensemble base, with 2 hierarchy structures of abstract representations of the ensemble patterns, extending from it.

In fact, that single node could reperesent the largest image part, where smaller feature parts would be placed closer to the ensemble and then have links extending to the larger body parts that they connect with. The structure realised by this image processing algorithm is therefore very close to that design. It is even possible to imagine that the smaller parts

doi:10.20944/preprints202201.0259.v1

would form first in a human brain - whether that is the ensemble pattern or the hierarchy representation of it, simply because they require fewer neurons, and then the largest part would form last.

3.4 Relative Positioning

The image parts can therefore be placed in order of their size and then what they link to. This can be done for each image individually, making it orthogonal and it is a very explainable process. Because the parts are easily recognised in the original image, their relative position can also be determined. For the current implementation, the full image is divided into 16 regions, where a 32x32 pixel image would be divided into 16 8x8 regions, for example. Each image part that links to another part has a positional array that stores a value for the 'North-South-West-East' directions. If the image part is positioned at the centre of the image it links with, then all of the values are 0. It can then either be 1 or 2 steps away in any of the 4 directions and this can be easily determined. This helps to put the image part into context with the larger part and when comparing images, the parts should have the same relation with their larger counterparts to be considered. Note that this does not require an exact positional match, but has some leeway as to where exactly the parts are placed. It would mean, for example, that an image with a line at the bottom would not be confused with an image with a line at the top, but two lines at the top do not have to be in exactly the same place. It also means that the image parts can be cropped before being compared, because their exact position is translated over to the positional array.

4 Implementation and Testing

A computer program has been written in the Java programming language. It is able to convert binary images into ascii 1-0 representations. These were then read into train and test datasets for each category. The train images were learned very quickly and most of the time was taken comparing the test images to the learned exemplars. For these small image sets however, processing a test image required only a few seconds. Then a count of the actual versus the closest match category for each test image was done, resulting in a percentage accuracy score.

4.1 Hand-Written Numbers Datasets

A first test used the Chars74K set of hand-written numbers [18], but only the numbers 1 to 9. There were approximately 55 examples of each number and the binary image was converted into a 32x32 black and white ascii image first. The examples were then divided into a train set of 40 images and a test set of 15 images. After exemplars were learned for the 9 train categories, each of the 15 images in the 9 test sets were compared and matched with their closest category. This was an exhaustive search process but very accurate, where each of the test set images was classified 100% correctly. As a comparison, an earlier image recognition attempt [11] only produced a 46% accuracy over the same dataset. The dataset was also used in [3], where they tested the full letters dataset, not just the numbers and produced possibly 55% accuracy. The Deep Learning methods however are able to recognise the number sets with an error percentage of only 1-2% (1.25%) [12], for example. A second test used the Semeion Handwritten Digits Dataset [1][17], with a split of 120 images in the train set and 40 images in the test set. The dataset contains 1593 handwritten digits from 0 to 9, converted into 16x16 black and white ascii images. The test images were again classified to 100% accuracy. The original paper [1] quoted a success score of about 93%, where mid-90% is quoted in other papers as well, and so 100% might be a new record.

5 Conclusions

This paper describes a new image-processing algorithm that is very human-like. It processes and stores images individually, but these can then be clustered into exemplars. The process uses an angular eye-scan that might have some biological reference and it is conjectured that the image parts are more 'intelligent' to what a neural network might produce and there is some direction to how it learns. The process can even include information about relative positions. The method is shown to be very quick for small image sets, but it requires an exhaustive search over all of the saved exemplars, which might require some sort of heuristic search, if the database was to grow very large. While the new method is probably quicker with the training phase alone, a neural network can store all of the images sets together, which should make it a lot quicker with the information retrieval. One other advantage is that because the image parts are clearly described and put into context, the whole process is very explainable.

References

[1] Buscema, M. (1998). MetaNet: The Theory of Independent Judges, in Substance Use & Misuse, Vol. 33, No. 2, pp. 439 - 461.

[2] Chen, K., Choi, H.J., and Bren, D.D. (2008). Visual Attention and Eye Movements.

[3] de Campos, T.E., Babu, B.R. and Varma, M. (2009). Character recognition in natural images, In Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP), Lisbon, Portugal.

[4] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K. and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, pp. 248-255.

[5] Fukishima, K. (1988). A Neural Network for Visual Pattern Recognition. IEEE Computer, 21(3), 65 - 75.

[6] Gatto, B.B., dos Santos, E.M., Fukui, K., Junior, W.S.S. and dos Santos, K.V. (2020). Fukunaga–Koontz Convolutional Network with Applications on Character Classification, Neural Processing Letters, Vol 52, pp. 443 - 465. https://doi.org/10.1007/s11063-020-10244-5.

[7] Greer, K. (2021). New Ideas for Brain Modelling 7, International Journal of Computational and Applied Mathematics & Computer Science, Volume 1, pp.34-45.

[8] Greer, K. (2021). Exemplars can Reciprocate Principal Components, *WSEAS Transactions on Computers*, ISSN / E-ISSN: 1109-2750 / 2224-2872, Volume 20, 2021, Art. #4, pp. 30-38.

[9] Greer, K. (2019). New Ideas for Brain Modelling 3, *Cognitive Systems Research*, Vol. 55, pp. 1-13, Elsevier. DOI: https://doi.org/10.1016/j.cogsys.2018.12.016.

[10] Greer, K. (2019). Image Recognition using Region Creep, available on arXiv at https://arxiv.org/abs/1909.10811.

[11] Greer, K. (2018). New Ideas for Brain Modelling 4, *BRAIN. Broad Research in Artificial Intelligence and Neuroscience*, Vol. 9, No. 2, pp. 155-167. ISSN 2067-3957. Also available on arXiv at https://arxiv.org/abs/1708.04806.

[12] Hinton, G.E., Osindero, S. and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets, Neural computation, Vol. 18, No. 7, pp. 1527 - 1554.

[13] Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pp. 1097-1105.

[14] Kuefler, A. (2016). Attentional Scene Classification with Human Eye Movements, http://cs231n.stanford.edu/reports/2016/pdfs/004_Report.pdf.

[15] LeCun, Y. (2015). What's Wrong with Deep Learning? In IEEE Conference on Computer Vision and Pattern Recognition.

[16] Rule, J.S. and Riesenhuber, M. (2021). Leveraging Prior Concept Learning Improves Generalization From Few Examples in Computational Models of Human Object Recognition, Frontiers in Computational Neuroscience, Vol. 14, Article 586671, doi: 10.3389/fncom.2020.586671.

[17] Semeion Research Center of Sciences of Communication, via Sersale 117, 00128 Rome, Italy, and Tattile Via Gaetano Donizetti, 1-3-5,25030 Mairano (Brescia), Italy.

[18] The Chars74K dataset, http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/. (last accessed 15/8/20).