Towards an Objective Assessment of Alzheimer's Disease: The Application of a Novel Evolutionary Algorithm in the Analysis of Figure Copying Tasks

Alex Hazell
The University of York
Department of Electronics
Heslington, York, YO10 5DD, UK
+44 1904 432351
ah514@york.ac.uk

Stephen L. Smith
The University of York
Department of Electronics
Heslington, York, YO10 5DD, UK
+44 1904 432351
sls5@ohm.york.ac.uk

ABSTRACT
Alzheimer's is a chronic debilitating neurodegenerative disease that is difficult to diagnose; conventional approaches are subjective and can be unreliable. This paper describes work towards an objective assessment that uses an evolutionary algorithm to assess an important symptom of the disease, the loss of visuo-spatial ability. Results are presented for application of the system in assessing the immature visuo-spatial ability of 7-11 year old children, which are used as a model for Alzheimer's disease patients.

Categories and Subject Descriptors
I.2.1 [**Artificial Intelligence**]: Applications and Expert Systems – *Medicine and science.* I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding – *Shape.*

General Terms
Algorithms, Measurement, Human Factors, Theory.

Keywords
Evolutionary algorithm, Cartesian genetic programming, medical applications, Alzheimer's disease, image analysis.

1. INTRODUCTION
Alzheimer's disease is a degenerative debilitating condition which mostly affects people in later life. It is difficult to diagnose and conventional approaches are subjective and thus, can be unreliable. This paper reports work towards developing an automated assessment of Alzheimer's disease that employs a novel evolutionary algorithm that can be conducted in the clinical environment and the doctor's surgery, using commonly available computing peripherals.

GECCO'08, July 12–16, 2008, Atlanta, Georgia, USA.

A common symptom of Alzheimer's disease is the loss of visuo-spatial ability, which can manifest itself in many every-day activities such as using a map or remembering the way out of a large building. It also affects the drawing of simple three-dimensional geometric shapes, such as a cube, and it is this that the system described in this paper is designed to assess.

Section 2 of this paper provides an introduction to Alzheimer's disease and describes a conventional method of subjectively assessing loss of visuo-spatial ability through distorted drawings of cubes. Section 3 describes the application of the evolutionary algorithm, a novel representation of Cartesian genetic programming. Experimental results of the algorithm applied to cube drawings exhibiting immature visuo-spatial ability, obtained from children as a model population of Alzheimer's patients, are described in Section 4 before conclusions are drawn in Section 5.

2. ALZHEIMER'S DISEASE
Alzheimer's disease (AD) is the most common form of dementia and accounts for 62% of dementia-type cases [8]. There are an estimated 24.3 million people in the world with dementia today and some 4.6 million new cases every year. The total number of effected people is expected to double every 20 years with the largest prevalence, an estimated 60%, occurring in developing countries [4].

The disease occurs when the amyloid β-protein forms miliary bodies (plaques) and dense bundles of fibrils (tangles) in the structure of the brain [1]. This leads to the disruption or death of neurons causing symptoms such as confusion, disorientation, memory loss and communication problems [19]. As a progressive disease, the symptoms become increasingly more severe as further parts of the brain are damaged and, consequently, the patient's condition will deteriorate.

The exact causes of the sporadic form of AD are unknown which is thought to be because of the complex interaction between genetic and environmental factors [1]. However, a strong correlation has been found between age and the occurrence of dementia with a prevalence of below 1% in individuals aged 60-64 and between 24% and 33% for individuals aged 85 and over in the western world [4]. Other suggested risk factors include low

mental ability in early life, reduced physical activity, brain trauma and occupational attainment [1].

2.1 Diagnosis of Alzheimer's disease

Whilst there is currently no cure for AD, early diagnosis is essential as it may permit treatment to slow down the progression of the disease and stabilize its symptoms. Medication can be prescribed to reverse non neuro-degeneration symptoms such as depression and it is also important to allow the patient time to prepare for the future.

Absolute diagnosis of AD is only possible by examining brain tissue and is therefore impractical whilst the patient is alive. Due to this difficulty, the determining the presence of AD is most often a diagnosis of exclusion, where the physician will try to find other causes of the symptoms often by using laboratory tests and neuroimaging techniques.

An important part of the diagnosis and monitoring of the disease is to perform a neurological examination to evaluate the extent of the impairment of the patient. The most common method of diagnosis based on these examinations is the NINCDS-ADRDA Alzheimer's Criteria [13] which examines eight cognitive domains: memory, language, perception, attention, constructive ability, orientation, problem solving and functional ability. Problems within these domains could suggest the onset of AD and the criteria leads to four possible outcomes: definite, probable, possible and unlikely Alzheimer's disease.

Geometric shape drawing tasks are often used as part of this assessment to evaluate visuo-spatial neglect. Several tests have been developed such as the Clock Drawing Test, the Rey-Osterrieth Complex Figure Test and cube drawing tests. Research into cube drawing ability has not only shown that it is a useful tool in the detection of AD but that it is also good at the detection of very mild AD [16]. For cube drawing assessments detailed marking criteria is used to grade the cube and hence determine the level of impairment. One example of such a criteria is presented in [2] which is used to mark the development of cube drawing ability of 7 to 10 year olds and shows many similarities with the criteria used in [16] to mark drawings of elderly and AD patients. The scoring system taken from [2] is as follows:

1. A single square or rectangle of any orientation
2. A set of interconnected squared or rectangles numbering more or less than the number of visible faces in the cube (three) or single trapezoid with some appropriate use of oblique lines.
3. A set of three interconnected squares or rectangles not appropriately arranged to represent the visible arrangement of faces in the cube or a set of interconnected squared or rectangles numbering more or less than the number of visible faces in the cube including some appropriate use of oblique lines.
4. A set of three interconnected squares or rectangles appropriately arranged to represent the visible arrangement of the faces of the cube or an inappropriately arranged set of three outlines including some appropriate use of oblique lines.
5. Drawings that show only visible faces of the cube appropriately arranged (as noted previously) and that reveal crude attempts to show depth through use of oblique lines, curvature, or modification to angles.
6. Drawings that approximate to oblique projection or linear perspective or drawings that approximate well to oblique projection or linear perspective but that are drawn to a horizontal ground line rather than to an oblique ground plane.
7. Drawings that are close approximations to oblique projection or linear perspective but that contain some inaccuracies in angular relations between lines.
8. Accurate portrayals of a cube in oblique projection or linear perspective.

Figure 1 taken from [2] shows eight example drawings which have been classified based on this system.

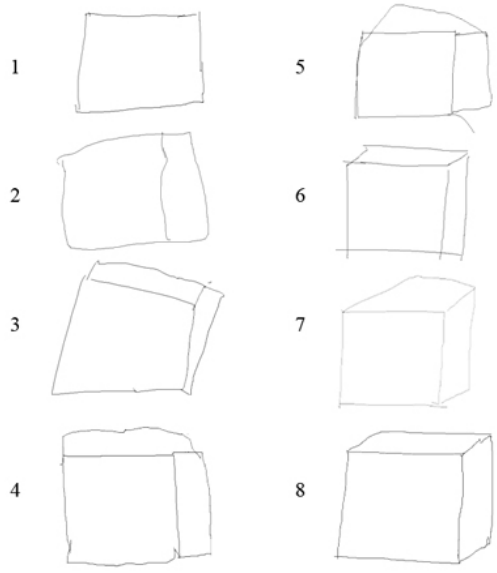

Figure 1. Eight classifications of the data used for this paper based on the marking system set out in [2].

Application of the assessment criteria by trained assessors can vary and hence, is arguably, unreliable, so it is desirable to produce an assessment mechanism which will be able to classify cube drawings in a completely objective way.

Guest and Fairhurst [6] implement an algorithm to extract components from static hand-drawn responses for two figure copying tests and one figure completion test. First the image is 'skeletonised' then split it into its horizontal, vertical and diagonal components by using directional neighborhood identification. The components are then assessed based on certain features are such as component omissions, length difference and spatial differences to examine the differences between neglect and control responses.

In [7] they extend this idea to include the analysis of dynamic performance features such as pen lifts, movement time and drawing time to improve the sensitivity of the assessment. By looking at these dynamic features they conclude that they can gain an additional understanding of the condition. However, the algorithms described in [6] and [7] use rigid sets of rules designed

by the authors based on observed differences. This paper proposes a method in which this level of subjectivity is removed.

3. APPLICATION OF EVOLUTIONARY ALGORITHM

3.1 Data acquisition

There are considerable practical difficulties in obtaining data from AD patients due to the lack of availability of patients willing or able to participate and the lengthy, but necessary, ethical approval procedures. However, as discussed earlier, there are close similarities between the development of visuo-spatial ability in 7 to 10 year old children and the respective loss of visuo-spatial ability observed in AD patients as the disease progresses. Therefore, children can provide a much easier way of obtaining data suitable for establishing this proof of concept study.

Drawings made by the children can be easily digitized by using a commercial digitizing tablet, such as that shown in Figure 2. The inking, wireless pen enables a traditional pen and paper environment to be preserved which reduces stress and distraction in the participants. Modern digitizing tablets can sample pen movements up to 200 times per second at a spatial resolution of up to 5000 lines per inch, enabling very fine reproduction of the drawings made.

Drawings were taken from children ranging from 7 to 11 years. Each child was asked to make several attempts at drawing a copy of a cube. Once the data was collected the cubes were manually classified using the scheme described by Bremner et al. in Section 2.1.

Figure 2. Commercial digitizing tablet (Photo: Wacom Europe GmbH).

3.2 Representation of data

The pre-processing stage uses raw data, obtained from the digitizing tablet as a stream of x-y coordinates charting the movement of the pen at fixed time intervals. The direction of the pen movements are calculated based on the angle between two sets of coordinates. In this study every twentieth set of x-y coordinates is taken from this data to reduce the effect of recorded noise that would have been introduced into the system if every increment between data points was used. The resulting the line is then assigned to one of four classes: horizontal lines ranging from -10° and +10° (measuring from the horizontal) were classified as 1, lines between 25° and 45° were classified as 2, vertical lines between 80° and 100° were classified as 3 and all other angles were classified as 0. Figure 3 shows these ranges of angles.

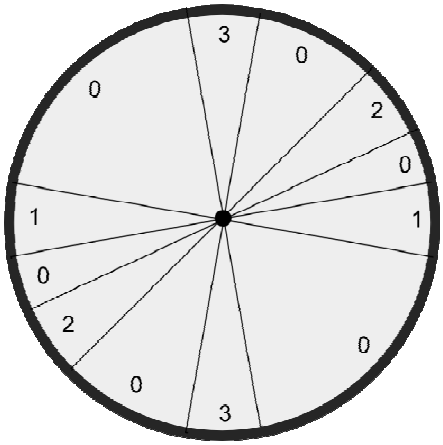

Figure 3. The range of classifications

The second group was deliberately not chosen to be half way between the vertical and horizontal boundaries but instead to favour the diagonal lines found in well drawn cubes where the typically correct angle appears to be around 30-40°. Whilst the limits of the horizontal and vertical lines might seem too lenient this is made to prevent the cubes, which are very well drawn but are drawn at a slight angle to the digitizing tablet are not unfairly classified as a patient response. This classification is saved to a file for subsequent use by the evolutionary algorithm.

3.3 Encoding the data

As the data was read into the algorithm there was an intermediate 'sub window' which encoded the classification into a three digit integer. The sub window read in a number of classifications and totaled the amount of each classification. It then normalized this so that each number was between 0 and 9, and put them together so that a triplet of integers referred to the relative amount of horizontal, oblique and vertical components present in each section of line. This number can be between 000 (for when there were no classifications of 1,2 or 3) and 900. Figure 4 shows three examples of the encoding.

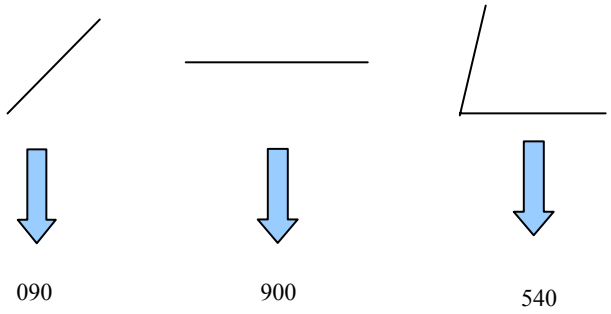

Figure 4. Three examples of the encoding scheme

The purpose of having an intermediate step between the preprocessing classification and the CGP is to preserve the

context of a line and to give the CGP a small overview of a particular section of the drawing. In this way the relationships between the different angled lines are considered by the algorithm rather than just the presence or absence of different angles. For example without the sub window the actually numeric value of a number would have little significance, but with the encoding a larger number signifies that the component makes up a larger part of a section of the drawing. This method has advantage of preserving the angular information in context but at the cost of reducing the amount of data the network uses.

3.4 Evolutionary algorithm

An implicit context representation of a Cartesian Genetic Program (CGP) was used for the evolutionary algorithm in this application.

Cartesian Genetic Programming (CGP) was first proposed by Miller [14][15] as an alternative representation for genetic programming which does not require the use of a parse-tree based programming language and does not exhibit uncontrolled expansion commonly termed bloat [9]. As opposed to the rigid tree structure representation of traditional GP, CGP permits the arrangement of nodes in a far more flexible, for example, rectangular format.

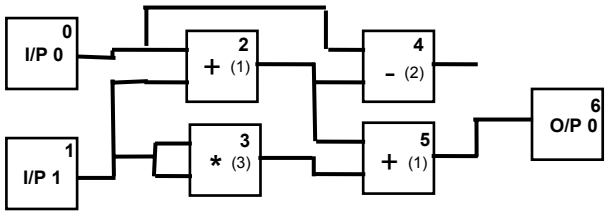

Figure 5. Example Cartesian Genetic Program. (Node number is specified in the top right hand corner of each node, the node function is specified in the centre with respective numeric reference in brackets.)

An example CGP network is shown in Figure 5 where four nodes are arranged in a 2 x 2 rectangular format. Two inputs, I/P 0 and I/P 1 convey the input values to the network and the result is presented at the output O/P 0. The structure each node simply consists of a function which manipulates the values presented at the inputs and sends the result to the output. A set of functions is available from which one is assigned to each node in the network. The nodes in the network are traditionally numbered consecutively starting at zero with the first of the input nodes, as shown in Figure 5.

The nodes within the architecture are configured by means of a *chromosome*, an example of which is given in Figure 6.

0 1 1 1 1 3 0 2 2 2 3 1 5

Figure 6. Example chromosome for configuration of the CGP network.

The chromosome consists of a string of integer values, arranged logically in groups of three, providing values for each respective (non-input) node in the network, i.e. the first triplet relates to node number 2, the second to the node number 3, and so on. The first two values of each triplet specify the nodes which are connected to the respective inputs and the third value the function to be applied to the values presented at the inputs.

A number of these chromosomes form the individuals of a population which are initialized with random values. Each chromosome is then used to configure the network to calculate a result for the problem under consideration. The result presented at the output of this network is compared with an ideal, and a fitness score derived, which is then associated with the respective individual's chromosome. After all the individuals in the population have been evaluated in this manner, the fittest is retained and used as the parent for a subsequent generation of individuals. These new individuals are generated by simply mutating the parent in a non-deterministic manner.

A criticism of CGP (and GP in general) is that the location of genes within the chromosome has a direct or indirect influence on the resulting phenotype [10]. In other words, the order in which specific information regarding the definition of the GP is stored has a direct or indirect effect on the operation, performance and characteristics of the resulting program. Such effects are considered undesirable as they may mask or modify the role of the specific genes in the generation of the phenotype (or resulting program). Consequently, GPs are often referred to as possessing a direct or indirect context representation.

An alternative representation for GPs in which genes do not express positional dependence has been proposed by Lones and Tyrrell [10][11][12]. Termed implicit context representation, the order in which genes are used to describe the phenotype (or resulting program) is determined after their self-organized binding, based on their own characteristics and not their specific location within the genotype. The result is an implicit context representation version of traditional parse-tree based GP termed Enzyme Genetic Programming. The authors have since implemented an implicit context representation of CGP, termed Implicit Context Representation Cartesian Genetic Programming (IRCGP), specifically for the evolution of image processing filters [18].

Implicit context representation employs an enzyme model comprising a shape (the component's output), activity (the component's function) and specificities (or binding sites – the component's inputs) [10], as shown in Figure 7. Along with input and output components the enzyme model can be considered a program component, executing one of the functions listed in Table 1, from which a genetic program may be constructed. The shape describes how the enzyme is seen by other program components. Similarly, the binding sites determine the shape (and hence type) of program component the enzyme wishes to bind to. Finally, the activity determines the logical function the enzyme is to perform. A typical IRCGP will comprise a set number of inputs and outputs and a number of enzyme models or components. Initial values for each component's binding sites and logical function are assigned non-deterministically; the component's shape, however, is computed by combining the numerical values of its binding sites' shapes and logical function as shown in Figure 8.

Once initialized, components are bound together to form a network, as shown in Figure 9. The order in which components are bound is determined by the least-squares match between a component's binding site shape and another component's shape.

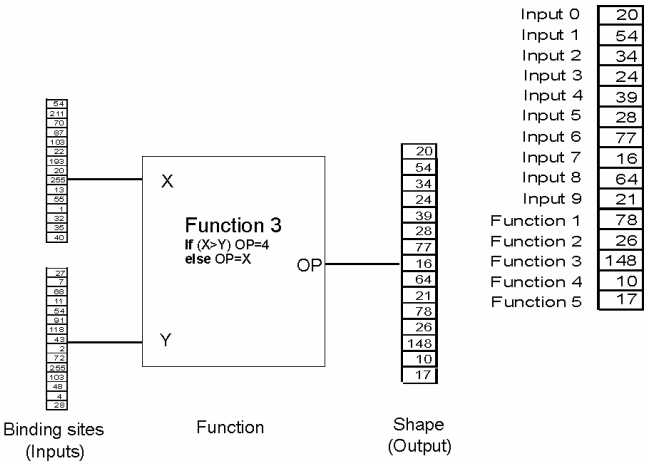

Figure 7. Example of a processing element that forms the evolutionary network.

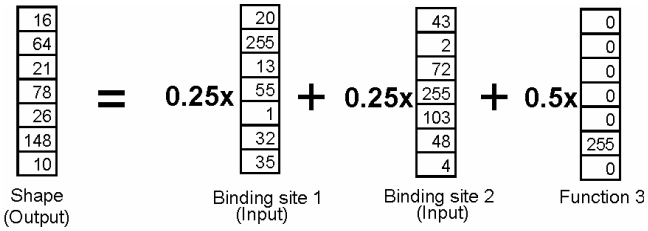

Figure 8. Derivation of a component's shape from the component's function and binding sites' shapes.

The best matching components are bound first and the process is repeated until a network has formed in which no further binding is possible.

Over time, components may evolve through mutation. Mutation is applied to the component's binding sites and logical function with a pre-determined probability. When this occurs, a new component shape is derived accordingly and may lead to different binding between components occurring. This in turn may result in a modified network.

The network of processing elements is arranged in 10 rows and 3 columns as shown in Figure 9. In addition, 10 input components and one output component can also be seen. The 10 input components are fed by the encoded classifications described in Section 3.2.

3.5 Function set

The function set designed for this application is novel in that each digit is processed separately to preserve the encoding. There are two inputs to each function and in this notation the two inputs are X which is encoded in the form A1, B1, C1 and Y which is encoded in the form A2, B2, C2.

The network has 7 functions available to it as shown in Table 1. Function 7 averages all 3 digits of the two encoded integer inputs (independently to each other) whilst the others compare one of the components from the two inputs (for example just the vertical component) and either outputs one of the inputs or averages the two. Function F7 averages each component individually without interfering with the other digits. The calculation is done by extracting each component using a mod function, averaging each and reconstructing the encoded integer. When the sum of the two components is odd the result is rounded up to retain a purely integer number and avoid interference between components.

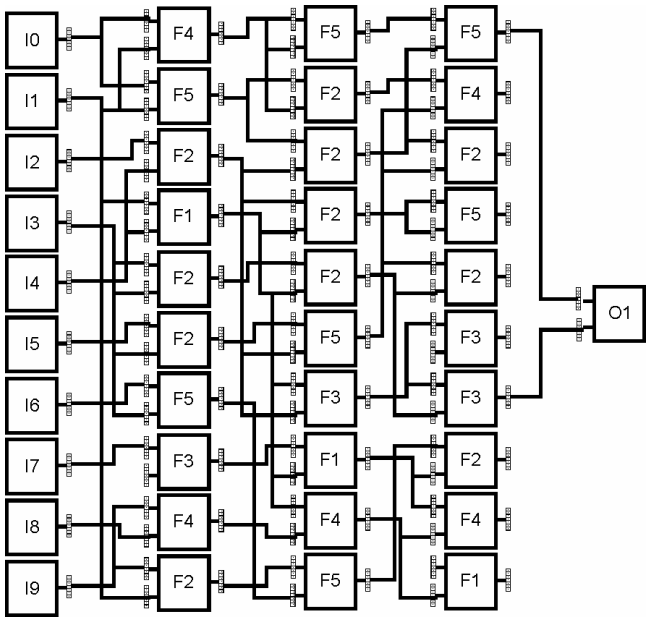

Figure 9. Example evolutionary network.

Table 1. Function set

Function Index	Function Definition
F1	**if** (A1>A2) OP= A1 **else** OP=average(X+Y)
F2	**if** (B1>B2) OP= B1 **else** OP=average(X+Y)
F3	**If** (C1>C2) OP= C1 **else** OP=average(X+Y)
F4	**if** (A2>A1) OP= A2 **else** OP=average(X+Y)
F5	**if** (B2>B1) OP= B2 **else** OP=average(X+Y)
F6	**if** (C2>C1) OP= C2 **else** OP=average(X+Y)
F7	OP = average(X+Y)

3.6 Fitness function

Two fitness functions were used to differentiate between control and patient responses by making comparisons between the different components at the output of the network. The first fitness function, shown below, is based on the principle that the horizontal and vertical content of patient and control files are different, with a patient response containing more vertical lines based on their diminished visuo-spatial ability in drawing oblique components. The second fitness function directly compares the amount of oblique component to the amount of horizontal component. By making a direct comparison between two

components (as opposed to making a comparison with a number) the effect of the difference in the size between different cube drawings is normalized.

```
if (loop==0) if (A < C)
pat_fitness++;
if (loop==1) if (A >= C)
con_fitness++;
```

Fitness function 1

```
if (loop==0) if (B*3 < A)
pat_fitness++;
if (loop==1) if (B*3 >= A)
con_fitness++;
```

Fitness function 2

3.7 Evolutionary parameters

For the results presented in this paper a network was evolved using a population size of 4 over 5000 generations. A conventional elitist strategy was adopted with a mutation rate of 6% for the function used by each component and 3% for each dimension of the binding sites' shapes.

4. RESULTS

4.1 Split of data

The data consisted of 142 drawings from 27 children. They were first arbitrarily split into training and testing sets, then split into patient and control groups so that cubes classified as 1-4 on Brenmer's scale were grouped as patient responses and cubes classified as 5-8 were grouped as control responses. For these results 22 patient and 51 control responses where used to train the algorithm and 21 patient and 48 control responses were used to test it.

After ten runs the highest evolved chromosome from the first fitness function had a patient fitness of 75% and control fitness of 94%. This chromosome was tested on the test set and the percentages of artifacts found are shown in Figure 10.

The second fitness function was then used to evolve a second network using the same training set, and after ten runs the highest evolved chromosome had a patient fitness of 69% and a control fitness of 79%. This chromosome was tested on the same test set and the percentage of artifacts detected are shown in Figure 11.

By setting a threshold of 50%, the first fitness function has 100% of patient responses below the threshold and 85% of the control responses above the threshold. The second fitness function has 86% of patient responses below the threshold and 98% of control responses above the threshold.

The results from the two fitness functions can be averaged to help minimize anomalies, the results of which are shown in Figure 12. For these combined results, setting a threshold of 60% results in 100% of the patient responses being below the threshold and 98% of control responses being above.

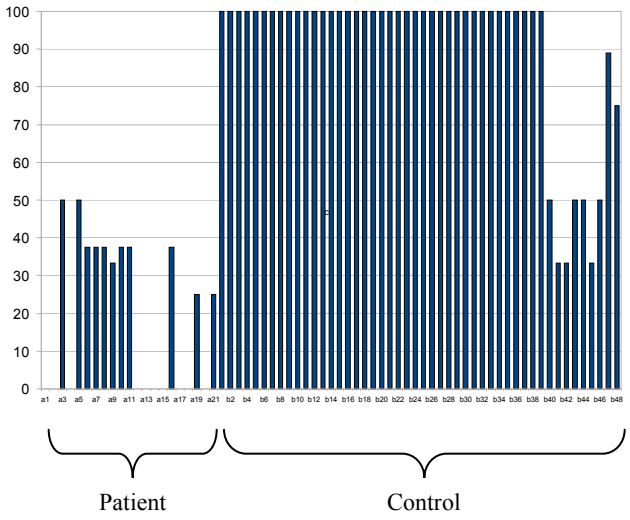

Figure 10. Percentages of artifacts in test responses for fitness function 1

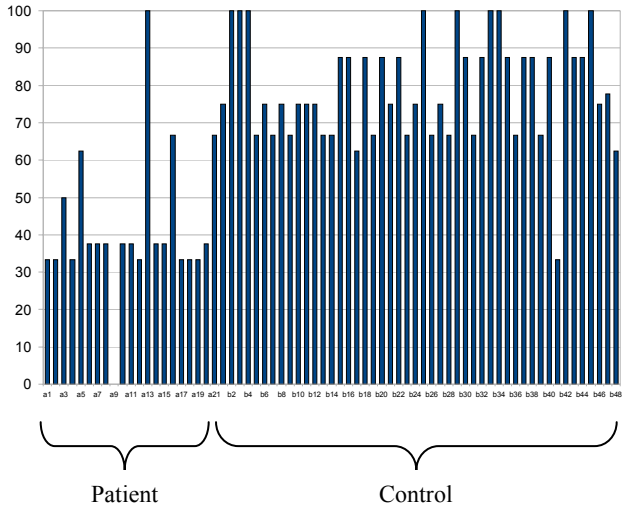

Figure 11. Percentages of artifacts in test responses for fitness function 2

5. CONCLUSIONS

This paper presents a novel system for the assessment of visuo-spatial ability – an important symptom of Alzheimer's disease.

The results show that a classification of subjects' drawings can be made using a special representation of Cartesian genetic programming (CGP) by taking the very basic angular information without the need for complex rule sets and image analysis. This accuracy might be improved further by extending the encoding scheme to include other properties in the data such as time taken, the rate of a change of angle and more complex features such as tremor or hesitation. It could also be possibly developed by using a different and maybe more extensive classification scheme.

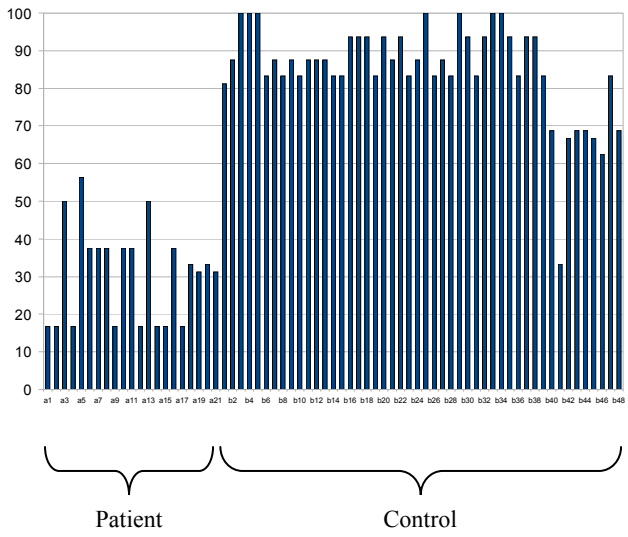

Figure 12. Percentages of artifacts in test responses for combined results of fitness function 1 and 2.

It is hoped that the method could be extended to include multiple classifications so that the algorithm could correctly and objectively group the data into the eight groups shown in Section 2.1 as a measure of the onset of the disease and as a diagnostic tool.

6. REFERENCES

[1] Blennow, K., Leon, M., Zetterberg, H., 2006. Alzheimer's disease, The Lancet, Volume 268, pp387-403.

[2] Bremner, J., Morse, R., Hughes, S., Andreasen, G. 2000. Relations between drawing cubes and copying line diagrams of cubes in 7-to 10- year old children, Child Development, Volume 71, Number 3, pp. 621-634.

[3] Dubois, B., Feldman, H., Jacova, C. 2007. Research criteria for the diagnosis of Alzheimer's disease: revising the NINCDS–ADRDA criteria. The Lancet Neurology , Volume 6 , Issue 8 , pp 734 - 746

[4] Ferri, C., Prince, M., Brayne, C. et al. 2006. Global prevalence of dementia: a Delphi consensus study. The Lancet, Volume 266, Issue 9503, pp 2112 – 2117.

[5] Gaestel, Y., Amieva, H., Letenneur, L., Dartigues, J., Fabrigoule, C. 2006. Cube Drawing Performances in Normal Ageing and Alzheimer's Disease: Data from the PAQUID Elderly Population-Based Cohort, Dementia and Geriatric Cognitive Disorders, 21, pp22-32

[6] Guest, R., Fairhurst, M., Potter, J., 2000. Automated Extraction of Image Segments from Clinically Diagnostic Hand-Drawn Geometric Shapes. Proceedings of the 26th Euromicro Conference, Volume 2, pp440 – 446.

[7] Guest, R., Fairhurst, M., 2002. A novel multi-stage approach to the detection of visuo-spatial neglect based on the analysis of figure-copying tasks. Proceedings of the fifth international ACM conference on Assistive technologies, pp157 – 161.

[8] Knapp, M., Prince, M. et al. 2007. Dementia UK: A report to the Alzheimer's Society on the prevalence and economic cost of dementia in the UK produced by King's College London and London School of Economics. Alzheimer's Society: London.

[9] Langdon, W. 2000. Quadratic bloat in genetic programming. Proceedings of the 2000 Genetic and Evolutionary Computation Conference, D. Whitley, D. Goldberg, and E. Cantu-Paz (Eds), pp. 451-458.

[10] Lones, M. A., 2003. Enzyme Genetic Programming. PhD Thesis, University of York, UK.

[11] Lones, M. A. and Tyrrell, A. M. 2002. Biomimetic representation with enzyme genetic programming. Genetic Programming and Evolvable Machines, vol. 3, no. 2, pp. 193-217.

[12] Lones, M. A. and Tyrrell, A. M. 2004. Modelling biological evolvability: implicit context and variation filtering in enzyme generic programming. BioSystems, vol. 76, no. 2, pp. 229-238.

[13] McKhann, G., Drachman, D., Folstein, M., Katzman, R., Price, D., Stadlan, EM. 1984. Clinical diagnosis of Alzheimer's disease: report of the NINCDS-ADRDA Work Group under the auspices of Department of Health and Human Services Task Force on Alzheimer's Disease, Neurology, Volume 34, pp939–44 .

[14] Miller, J.F. and Thomson, P. 2000. Cartesian Genetic Programming. Third European Conf. Genetic Programming, R. Poli, W. Banzhaf, W. B. Langdon, J. F. Miller, P. Nordin, and T. C. Fogarty (Eds.), Lecture Notes in Computer Science, vol. 1802.

[15] Miller, J.F., Job, D. and Vasilev, V. K. 2000. Principles in the evolutionary design of digital circuits—Part I. Genetic Programming and Evolvable Machines, vol. 1, pp. 7–36.

[16] Shimada, Y., Meguro, K., Kasai, M. et al. 2006. Necker cube copying ability in normal elderly and Alzheimer's disease. A community-based study: The Tajiri project, Psychogeriatrics, volume 6, pp4-9

[17] Smith, S.L., Gaughan, P., Halliday, D. et al. 2007. Diagnosis of Parkinson's Disease using Evolutionary Algorithms, Genetic Programming and Evolvable Machines, Volume 8, Issue 4, pp433-447.

[18] Smith, S.L., Leggett, S. and Tyrrell, A. M., 2005. An implicit context representation for evolving image processing filters. Proceedings of the 7th Workshop on Evolutionary Computation in Image Analysis and Signal Processing, Lecture Notes in Computer Science, vol. 3449, pp. 407-416.

[19] Wolfson, W. 2008. Unraveling the Tangled Brain of Alzheimer's. Chemistry & Biology , Volume 15 , Issue 2 , pp 89 - 90.