

THE EVALUATION OF IMAGE QUALITY FOR TRAINING SIMULATORS USING ARTIFICIAL INTELLIGENCE

R.J. Cant and A.R. Cook,

Real Time Telemetry Group, Department of Computing,

The Nottingham Trent University, Burton Street, Nottingham NG1 4BU. U.K.

Tel. +44-115-9418418, Fax +44-115-9486518. E-mail rcc@doc.ntu.ac.uk

ABSTRACT

A method of determining the effectiveness for training of a visual or sensor (for example radar or sonar) simulation system is presented. The method uses a neural network to perform the evaluation. This avoids the cost and delay associated with an experiment which uses human subjects. A specially devised neural network architecture is described and the use of the system for two example studies is presented. The first study concerns the effect of resolution and anti-aliasing and it is shown that whilst the network can learn to distinguish very low resolution images, if anti-aliasing is not used then the training does not transfer back to the higher resolution case. In the second study we compare Phong and Gouraud shading and show that the small theoretical advantage which Phong shading has in the representation of lighting angle can be detected by our method.

INTRODUCTION

Simulators have been used to train military personnel for many years and their use is now expanding even more rapidly as budgets become more constrained. At the same time, cost constraints for the simulators themselves are becoming ever tighter yet the accuracy of the simulation must improve to allow them to be used in more sophisticated ways.

To reconcile these contradictory requirements it is necessary to tune the performance of the simulator to the specific application (Cant and Sherlock 1987; Cant 1994). This allows the cues which are critical for training to be provided whilst false cues and unnecessary expense are avoided. To do this we must know accurately which features are important in each part of the simulator-trainee interface.

The most critical parts of this interface are likely to be the graphic displays which are used to present visual and other sensor (infra-red, sonar, radar, etc) information. In the past the development of computer generated imagery has largely been driven by subjective criteria of realism, whilst requirements for radar and sonar simulation have often been derived using little more than guesswork.

The obvious way to improve on this situation would be to set up an experiment using a group of human volunteers. In order to be reliable this would necessarily be a relatively slow process and would require equipment of cost comparable to that of the final product. It might also need to use trained personnel as guinea pigs.

None of these factors is particularly problematic for an academic study but in the commercial world the situation is different. Such a procedure does not fit into a typical training simulator lifecycle. Initial design decisions are usually made during a competitive

tendering process with timescales of about a month and small teams working with limited resources. A human based study would be practical if it could be done just once and the results used for all subsequent applications. In practice however, there is a steady stream of new applications and techniques that would each require a new study to be done.

The variations between one simulator and another also multiply the number of studies which are required. Even within a single application area, superficially similar simulators may be used for different tasks. For example, the visual simulation requirements for a flight simulator which is being used to train pilots in "nap of the earth" flying will be very different from those for a trainer which is being used for air combat at higher altitudes, even if the aircraft is the same.

There is also a need to verify the performance of equipment when it is delivered and when any modifications are made. The databases describing the landscape and vehicles to be displayed will also need to be individually tested (and re-tested when they are modified). Once again a human based study is impractical because it would have to be undertaken at a very busy stage of the project with limited time available and limited access to equipment and personnel.

What is required is a method of assessment which is objective, can be undertaken by one or two people, requires no special equipment and can be completed in a month or two. It should also be possible to relate the analysis directly to the tasks which are important for each individual simulator.

The mode of operation which we adopt is to generate a set of images, typical of the images which will be generated in the simulator, and present it to an analysis program together with some data which defines the training task. The analysis program should then feed back results which indicate how useful the images are for training. It is a critical advantage of this technique that the images do not have to be generated in real time, even though the simulator will require real time interaction, thus allowing the images to be generated using software on any general purpose computing system that happens to be available.

The core of the analysis program is a neural network to which sample images are presented together with the correct output information. Initially this information is used to train the network. Later it can be used to test how well the network has learnt. By comparing the learning capability of a fixed network configuration for different qualities of images it is possible to obtain information about how well the image generation algorithms are performing.

The range of possible applications for this technique is wide and the most valuable ones are likely to be in relatively obscure areas such as thermal imaging or sonar simulation where few people have enough experience to provide intuitive information. For initial testing of the technique we have chosen visual simulation precisely because there are many areas in which one intuitively has a clear idea of what the results ought to be.

The possible training tasks include object detection, recognition, determination of object attitude, distance, or speed and determination of own position and movement from background visual cues. Particular training simulators will have their own special requirements so this list is far from exhaustive. The possible image generator quality features which can be judged include resolution, sampling (anti-aliasing) techniques, rendering algorithms, database complexity and any special purpose algorithms devised for the particular system.

TRAINING CONSIDERATIONS

In judging the suitability of an image for training one is looking for three things:

- (i) sufficient information to perform the training task;
- (ii) no extraneous information which could distract a trainee;
- (iii) correspondence between the way the information is presented in the training image and the way it is presented in the real world.

Failures in each of these three categories will have different effects on the training utility of the image. A failure in category (i) will not only destroy the usefulness of the trainer for the given task but will cause knock on effects on any tasks which are dependent upon it. A category (ii) failure is likely to result in fatigue when the simulator is used for long periods and may also cause negative training. A problem of the kind described in (iii) above will destroy the usefulness of the trainer for the given task and may also cause some negative training but will not interfere with training for dependent tasks.

In many cases it is possible to make an assessment of the performance of an image generator directly from information about resolution and rendering algorithms. For example, given the resolution, sampling algorithm and field of view then one can calculate the range at which an object of a given size will be visible. This can then be compared with the performance of the human eye in the equivalent real world situation. However, most training tasks are subtler than this and, given the complexity of most model databases, the computations involved are likely to become intractable. A more sophisticated model of human perception would also be required to achieve an accurate result. We need a general purpose model that can be driven from the outside and will adapt itself to the task presented to it. A neural network is well suited to these requirements. Neural networks have been shown mathematically to be universal approximators, capable of solving any problem (Hornik et al. 1989). They have also had a considerable amount of success in pattern recognition, which is similar to the tasks which we are concerned with here.

A neural network will not automatically perform to exactly the same level as a human, so the results cannot be taken simply at face value, but with some care it is possible to extract accurate information from the system. To understand what is necessary to obtain such results, we examine in turn each of the three categories defined above.

For category (i) it is clear that, if the network can be successfully trained for the task then there must be sufficient information in the image. A negative result does not guarantee that the reverse is true but, because the limitations of the network are qualitatively different from those of the image, we have been able to separate them by observing the effect of changes in the network design, the learning mechanism or the set of images employed.

In category (ii) the outcome will depend on the type of network that is employed. If only supervised learning is used then it is unlikely that the network will detect image defects, unless they are very dominant or correlate closely with the output categories. If on the other hand a network with an unsupervised hidden layer is used, then it will classify images according to their most significant features irrespective of whether the resulting categories correspond with its training objectives. It should then be possible to detect whether the network has "seen" the features by examining the weights in the hidden layer after training.

The final category requires the most careful analysis because the artificial neural network will not necessarily mimic the human subject in its perception. A mismatch of this kind could cause either a false negative or a false positive result. The false negative will cause a category (i) type failure which we have already analysed above. The false positive is effectively a kind of negative training specific to the neural network. This could arise for one of two reasons. Firstly, because the neural network lacks the extra domain knowledge that a human trainee would have, it is likely to pay attention to things which the human being would ignore. This can be checked by using a comparison set of "real" images which would not contain the misleading information in the same way. Secondly, because the neural network works from a numerical representation of the image, it may be able to use criteria which are not available to a human. Once again a comparison with real images may be useful but a better solution to this problem is to perform a study using human subjects in parallel with the neural network. At first sight this might seem like a negation of our approach, however this study need not be performed afresh for every single case. It need only be done for a representative sample. Given reasonably consistent results it should be possible to generalise them by tuning the network design. Alternatively a set of rules for the interpretation of network results could be produced. We intend to perform such a study in the future. At present we are exploring the capabilities of the neural network to distinguish different image generation algorithms and determine the effects of image resolution.

In the present work we are attempting to model human behaviour rather than achieve optimum network performance. We must therefore pay some attention to the way in which humans typically learn. Initially a familiarisation phase is necessary during which a

person is exposed to the environment which they need to learn about. At this stage explicit instruction is not very useful because of the lack of experience. Later on, when the situation is more familiar, supervision can be employed to improve performance. We will attempt to follow this sequence in the neural net architecture outlined below. To relate the network performance to human performance it is also desirable to understand the reasons why the system behaves as it does.

NEURAL NETWORK ARCHITECTURE

Back error propagation has been the most popular network model in recent years and is theoretically capable of the highest performance. It does, however, suffer from three critical problems in the present case. It learns slowly, can converge to poor solutions (local minima), and leads to a solution which can be difficult to understand. In the present context there is an additional problem with supervised learning in that the network is likely to ignore image defects more readily than a human subject would.

Counter propagation networks are simpler to understand since they work by matching the pattern of weights in the hidden layer to the input image as a whole. This type of learning also produces rapid convergence, especially if a large neighbourhood is used early in the training process as in the Kohonen neural network. The major deficiency of counter propagation is that, in its pure form, its ultimate performance is limited unless a very large hidden layer is used, allowing it to simply learn all possible input patterns. We believe that this mode of operation is only partially representative of human learning but provides a good starting point from which to work. We also need to obtain good performance in order that we can look at the characteristics of the input images rather than those of the network itself.

The usual way to improve performance is to apply a supervising influence back into the hidden layer from the output layer. In the basic counter propagation model each weight vector in the hidden layer is an exemplar of a set of similar images which appeared in the training set. There is no guarantee that it corresponds directly to one of the output classifications. This is a particular problem in the present case because we will often be constructing output categories as arbitrary subdivisions of a continuous space rather than a discrete set of possibilities as would occur in character recognition.

By supervising the hidden layer we can encourage it to contain exemplars of the output categories, or, even better, discriminators optimised to distinguish between the categories. There are a number of ways in which this can be done. In the early stages of the present work a relatively arbitrary method was followed, (Williamson 1994) which was quite similar to some of the learning vector quantisation techniques described in (Kohonen 1990). We have now progressed to a more systematic method which allows back propagation learning techniques to be implemented as a tuning mechanism within a counter propagation architecture.

The key to this method is the design of the hidden layer and its connection to the output layer. In counter propagation architectures there are two ways of determining the error on the hidden layer. The first method is to normalise both the weights and the input pattern, allowing the dot product to be used to measure the

similarity between the two. A normalisation error is obtained when this dot product has its maximum value of 1. Unfortunately the normalisation process prevents back error propagation corrections from being calculated efficiently so this method cannot be used for our purposes. The alternative is to measure error as the Euclidean distance between the ends of the input and hidden vectors. This gives a zero value when there is an exact match. This value cannot be used directly as an activation, so usually it is used only to identify the "hidden winner" with the actual values being ignored. Our innovation is to use the value:

$$a_n = e^{-\epsilon_n^2 / \sigma^2} \quad (1)$$

(Where

$$\epsilon_n^2 = \sum_k (w_{kn} - s_k)^2, \quad (2)$$

the w_{kn} are the weights on the hidden layer, the s_k are the input image and σ is a normalisation factor.)

as the activation on the n th hidden node. These activations can then be used to calculate output activations A_m in a manner similar to a back propagation network, although without a threshold function on the output nodes. (The exponential acts as a threshold function for the hidden layer.)

$$A_m = \sum_n a_n W_{nm} \quad (3)$$

The overall error is given by:

$$E = \sum_m (A_m - D_m)^2 \quad (4)$$

where D_m is the desired value of A_m .

We can now use the standard steepest descent method to calculate a correction on both the output and hidden layers:

$$\Delta W_{nm} = -\lambda (A_m - D_m) \cdot a_n \quad (5)$$

$$\Delta w_{kn} = \frac{-\lambda}{\sigma^2} a_n (w_{kn} - s_k) \left[\sum_m (A_m - D_m) W_{nm} \right] \quad (6)$$

where λ is a learning rate variable.

Notice that the choice of the normalisation factor σ is critical to the system. One might expect that this network could be used as a standard back propagation network using the equations above, but this does not work. The reason is that if a value of σ is chosen which will ultimately be suitable for a trained network then the a_n 's will initially be exponentially small and no learning will take place. If on the other hand a larger value is chosen then later all the a_n 's will be close to 1 and the network will not work properly. There are two solutions to this problem. One possibility is to gradually reduce the value of σ as training progresses. This approach works but is rather slow. The method we prefer is to train the hidden layer initially with unsupervised methods as described in (Kohonen 1990). When typical hidden errors have stabilised a value for σ can be chosen (a value slightly larger than the average "winning" hidden layer error works well). The output layer can now be trained using equation (5) above. When performance improvements have stopped we have an opportunity to assess the

results of unsupervised learning alone, which may in some cases turn out to be more representative of human performance than supervised learning. The next phase is to use equation (6) to optimise the hidden layer. This typically results in a substantial performance improvement compared to the unsupervised network. It is interesting to note that each hidden node contains a whole image pattern rather than simply a single feature as often happens with conventional back propagation networks. This remains true even when only back propagation training has been used because equations (1) and (2) above guarantee it. This method also "remembers" the initial unsupervised learning process because each of the hidden node images has been derived by modifying a pattern that was previously learnt. As a result of this the later optimisation can sometimes be rather slow if there has been some deficiency in the initial unsupervised phase of learning.

RESULTS AND ANALYSIS

In this paper we present two studies. In the first we analyse the effect of resolution and sampling techniques on the determination of object attitude. In the second we make a comparison between the Gouraud and Phong shading algorithms based on the degree to which the illumination direction can be determined from the image.

In each case the neural networks were initially trained competitively with a two dimensional neighbourhood in the hidden layer. This was done to match the dimensionality of the output categories.

Resolution and Sampling

Many training simulators are performance limited by the resolution of their displays. Examples include anti-aircraft missile simulators, air traffic control simulators and periscope simulators. The capabilities of the eye are difficult if not impossible to match using video technology. Typically a display having over 5000 lines is required to match the eye's resolving power. Even with a display of such high resolution, aliasing phenomena would be detectable if the image was not properly sampled. Since it is not possible to match the eye's performance economically, training programmes must be designed around the capabilities of the simulator as they are. Typical training tasks which are likely to test the resolution of the system are vehical detection, recognition and determination of attitude at long range. All these tasks will be affected both by raw resolution and by the accuracy of the sampling process.

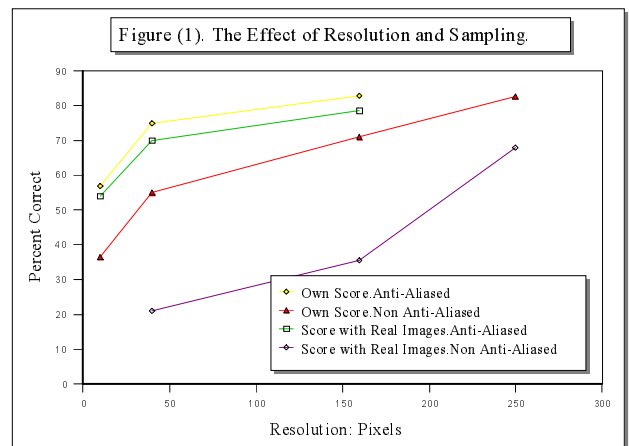
In our first study we have attempted to determine the limiting resolution at which attitude can be determined reliably. Since this has been done with and without antialiasing we have been able to analyse the tradeoff between increasing the resolution and applying antialiasing. Three models of fighter aircraft were used, a Harrier, a Mig 27 and an F16. The images presented to the network were selected randomly over the full range of angles in yaw and with a variation in roll of up to 15 degrees in either direction.

For each test a network of the type described above was trained to determine which aircraft was in the picture and determine its heading to the nearest 45 degrees. The network had a hidden layer of 192 neurons in a 16x12 array. The networks were trained continually until further improvements were insignificant. In all cases at least 5000 images were presented in the initial competitive

propagation mode. A minimum of 40000 in the second back

Figure (1) shows the percentage of correct classifications as a function of resolution (expressed in terms of the size of the aircraft image in pixels) for antialiased and non-antialiased images. From this data it is clear that the network is very good at the task, (perhaps too good) even without antialiasing.

To clarify what is going on we scaled the trained networks up by replicating the input nodes so that they could be presented with higher resolution ("real") images as a comparison. The result of this procedure was most revealing at the 10x4 resolution. Whereas the network that had been trained with anti-aliased images achieved almost the same score as it had with its "training set" (70% as opposed to 75%), the network which had trained on non-antialiased images performed much worse (21%, down from 55%). Clearly the training which the second network received was less relevant to the true task than that of the first. These results are also summarised in Figure (1).



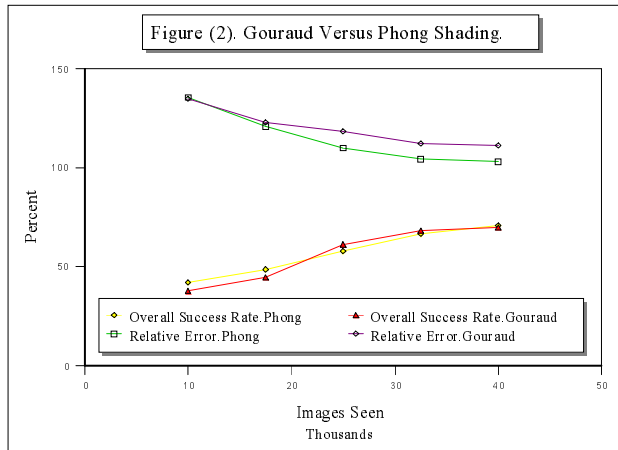
Shading Algorithms

A number of shading algorithms have been developed to allow curved surfaces to be represented realistically in computer generated images. Only the simpler methods have so far been applied to real time systems such as are used in visual simulation. The utility of such algorithms in training simulators is largely cosmetic but there is a possibility that, by providing an accurate cue about the lighting direction they could assist indirectly in the determination of object attitude.

To test this theory we set up a static image of the classic "chalice" model, used in many computer graphics demonstrations in the past. We varied the lighting direction and trained the network to recognise where the light was coming from. The azimuth orientation of the model was varied randomly to prevent the network from simply learning the possibilities by rote. There were eight categories in azimuth and three in elevation. The hidden layer had 96 neurons arranged in a 16x6 array. The images were initially generated at 150x150 resolution and then averaged down to 50x50 before being presented to the network. The shading algorithms used were Gouraud shading and Phong shading. A mixture of direct and ambient lighting was used, together with a mixture of specular and diffuse reflection. In theory Phong shading should have an

advantage over Gouraud because it allows the position of a highlight to vary continuously instead of being confined to the edges and vertices of polygons.

In Figure (2) we have plotted the progress of the training of the two networks. The lower curves show the overall percentage of correct categorizations. In the upper curves we have isolated the error in azimuth angle and expressed the result as a percentage of the inevitable error which arises from the discrete 45 degree categories. It is interesting that the Gouraud shading system actually slightly outperforms the Phong one in the vertical direction but is substantially worse in terms of ability to accurately judge the azimuth angle.



CONCLUSIONS

We have shown that neural networks are capable of deriving useful indications of the relative merits of different image generation algorithms. We hope to calibrate these results soon by performing a parallel study of human performance.

There is also considerable scope for further work to explore the implications of different databases, algorithms and training tasks. It will be interesting in future to analyse a real training simulator requirement to see how the design could be optimised using this technique.

Given more experience it should be possible for assessment of the image quality needed for a training simulator to be done quickly, economically and without subjective factors. It may also be possible to use the idea in other virtual reality applications. The leisure industry (which largely parallels the training simulator industry) provides an obvious example. It may also be possible to tune a "live" computer interface by testing the presentation of information in this way. A related application is the use of a virtual training simulator, for the training of robots which use neural network based control systems (Kuhn et al 1994).

Acknowledgements

The authors would like to acknowledge the contribution of Mark Williamson, whose final year degree project proved the feasibility of the current work and provided a starting point for it. We would also like to thank Marconi Simulation and Training (formerly

Perrault Simulation and Training) of Stockport England for the provision of some of the model databases which were used in our tests.

REFERENCES

- Cant, R.J. and P.E. Sherlock. 1987. "CIG System for Periscope Observer Training." In *Proceedings of the 9th Interservice/Industry Training Systems Conference*, 311-314.
- Cant R.J. 1994. "A Flexible Approach to Parallel, Real Time Graphics." In *Proceedings of the Second International Conference on Software for Multiprocessors and Supercomputers Theory, Practice and Experience* (Moscow, Sept.19-23). Russian Academy of Sciences, 179-188.
- Hornik, K ; M. Stinchcombe; and H. White. 1989 . "Multilayer Feedforward Networks are Universal Approximators." In *Neural Networks*, Vol2, Pergamon Press Ltd., Oxford, England. 359-366.
- Kohonen, T. 1990. "The Self Organising Map." *Proceedings of the IEEE*, 78, No. 9 (Sept) : 1464-1480.
- Kuhn, D.; J.P.Urban; S.Hagmann; and H. Kihl. "A Transputer Based Robot Vision System to Evaluate Neuro-Control." In *Proceedings of the Second International Conference on Software for Multiprocessors and Supercomputers Theory, Practice and Experience* (Moscow, Sept.19-23). Russian Academy of Sciences, 144-150.
- Williamson, M.V. 1994."Evaluation of the Effectiveness of Computer-Generated Images using Computer-Based Image Recognition." Project Report, The Nottingham Trent University, Burton St Nottingham U.K.