Abstract—In this work, we show that both fine-tune learning and cross-domain sim-to-real transfer learning from virtual to real-world environments improve the starting and final scene classification abilities of a computer vision model. A 6-class computer vision problem of scene classification is presented from both videogame environments and photographs of the real world, where both datasets have the same classes. 12 networks are trained from 2, 4, 8, ..., 4096 hidden interpretation neurons following a fine-tuned VGG16 Convolutional Neural Network for a dataset of virtual data gathered from the Unity game engine and for a photographic dataset gathered from an online image search engine. 12 Transfer Learning networks are then benchmarked using the trained networks on virtual data as a starting weight distribution for a neural network to classify the real-world dataset. Results show that all of the transfer networks have a higher starting accuracy pre-training, with the best showing an improvement of +48.34% image classification ability and an average increase of +38.33% for the starting abilities of all hyperparameter sets benchmarked. Of the 12 experiments, nine transfer experiments showed an improvement over non-transfer learning, two showed a slightly lower ability, and one did not change. The best accuracy overall was obtained by a transfer learning model with a layer of 64 interpretation neurons scoring 89.16% compared to the non-transfer counterpart of 88.27%. An average increase of +7.15% was observed over all experiments. The main finding is that not only can a higher final classification accuracy be achieved, but strong classification abilities prior to any training whatsoever are also encountered when transferring knowledge from simulation to real-world data, proving useful domain knowledge transfer between the datasets.

Keywords—Sim-to-real, Transfer Learning, Deep Learning, Computer Vision, Autonomous Perception, Scene Classification, Environment Recognition

I. INTRODUCTION

The possibility of transfer learning from simulated data to real-world application is promising due to the scarcity of real-world labelled data being an issue encountered in many applications of machine learning and artificial intelligence [1], [2], [3]. Based on this, Fine-tune Learning and Transfer learning are often both considered to be viable solutions to the issue of data scarcity in the scientific state-of-the-art via large-scale models such as ImageNet and VGG16 for the former and methods such as rule and weight transfer for the latter [4], [5], [6]. Here, we attempt to perform both of these methods in a pipeline for scene classification, by fine-tuning a large-scale model and transferring knowledge between rules learnt from simulation to real-world datasets.

The consumer-level quality of videogame technology has rapidly improved towards arguable photo-realistic graphical quality through ray-traced lighting, high resolution photographic textures and Physically Based Rendering (PBR) to name but several prominent techniques. This then raises the question, since simulated environments are ever more realistic, is it possible to transfer knowledge from them to real-world situations? Should this be possible, the problem of data scarcity would be mitigated, and also a more optimal process of learning would become possible by introducing a starting point learned from simulation. If this process provides a better starting point than, for example, a classical random weight distribution, then fewer computational resources are required to learn about the real world and also fewer labelled data points are required. In addition, if this process is improved further, learning from real-world data may not actually be required at all.

In this work, we perform 12 individual topology experiments in order to show that real-world classification of relatively scarce data can be improved via pre-training said models on simulation data from a high-quality videogame environment. The weights developed on simulation data are applied as a starting point for the backpropagation learning of real-world data, and we find that both starting accuracies and asymptotes (final ability) are often higher when the model has been able to train on simulation data before considering real data.

The main scientific contributions of this work are threefold:
1) The formation of two datasets for a 6-class scene classification dataset, both artificial simulation and real-world photographic data
2) 24 topology tuning experiments for best classification of the two datasets, 12 for each of the datasets by 2, 4, 8...4096 interpretation neurons following the fine

tuning of a VGG16 CNN network (with interpretation and softmax layers removed). This provides a baseline comparison for Transfer Learning as well as the pre-trained weights to be used in the following experiment.

3) 12 transfer learning experiments of the weights trained on simulation data transferred to networks with the task of classifying real-world data. The results are evidence that transfer learning of useful domain knowledge is possible from the classification of simulated environments to the classification of real-world photographic data, further improving classification ability of real data.

The remainder of this article is organised as follows: in Section II the state of the art in knowledge transfer from virtual worlds to real world is discussed, in Section III our methodology is outlined, while in Section IV experimental results are presented and analysed. A discussion of possible future work is provided in Section V before a final conclusion to this study is drawn in Section VI.

II. BACKGROUND AND RELATED WORK

In this section, state of the art in the area of knowledge transfer from virtual world tasks to real life tasks is discussed. The possibility of transfer from modern videogames to reality for complex problems is a new and rapidly growing line of thought within the field of deep learning. Related works are limited due to the young age of the field\(^2\).

Technologies such as realistic Ray Tracing and PBR in conjunction with photorealistic or photographically-enhanced textures enable photorealism in simulated environments (in this context, generated as a videogame environment). Ray Tracing is a rendering technique that works by following the individual pixel paths of light and simulating its physical properties when interacting with objects in the scene, which produces higher levels of realism in terms of lighting as opposed to the classical row-by-row scanline method [8]. Following various methods of implementation – Pharr et al. provide a detailed review of PBR methods [9]– PBR is the concept of combining high quality 3D models and surface-measured shading in order to produce accurate representations of real materials and thus photo-realistic quality objects [10]. An example of the quality of simulation possible through the usage of these technologies can be seen in Figure 1, developed by ArchVizPRO [7].

Transfer Learning is the improvement of a learning process for a new task by transferring knowledge from a related so-called source task to the new task, which is called target task. In this study, trained weights from one classification problem are used as the initial weights for a second problem and are subsequently compared to standard random weight distribution for this same problem [11]. The issue of data availability is recognised in a notable survey on transfer learning, where transfer learning approaches are suggested to produce better solutions for a second task characterised by more limited data than the first task [12]. The reduced availability of real-world data in comparison to the almost infinite possibilities in virtual environments is such a scenario.

Kim and Park recently argued against a classical heuristic search approach for the tracking of road lanes in favour of a deep learning approach featuring transfer learning from Grand Theft Auto V (GTA V) and TORCS environments [13]. GTA V was also used to gather data for a computer vision experiment in which vehicle collisions were successfully predicted when transfer learning was applied [14]. Trial-and-error learning is not suited to high-risk activities such as driving, and so, reinforcement learning is not possible when the starting point is a real-world situation; researchers argue that transfer of knowledge can improve the ability to perform complex tasks, when initially performed in simulation [15] and [16]. For autonomous navigation, environment mapping and recognition is a very important task for self-driving vehicles, many of which consider LiDAR data as input towards mapping and subsequent successful real-time navigation [17], [18], [19], [20].

In addition to LiDAR, many authors have argued for the processing of photographic image data for environment or scene recognition. Herranz et al. [21] show that classification of both scenes and objects reach human-level classification abilities of 70.17% on the SUN397 places dataset via manually chosen combinations of ImageNet-CNNs and Places-CNNs. Similarly, Wu et al. [22] achieved accuracy of 58.11% on the same dataset through harvesting discriminative meta-objects, outperforming Places-CNN (AlexNet fine tuning), which had a benchmark accuracy of 56.2% [23].

In Tobin et al. researchers trained computer vision models with a technique of domain randomisation for object recogni-
tion within a real-world simulation, which, when transferred to real-world data, could recognise objects within an error of around 1.5 centimetres [24]. This was further improved when it was noted that a dataset of synthetic images from a virtual environment could be used to train a real-world computer vision model within an error rate of 1.5 to 3.5 millimetres on average [25]. Researchers noted that virtual environments were simply treated as simply another variation rather than providing unsuitable noise. Similarly, computer vision models were also improved when initially trained in simulation for further application in reality where distance error rates were reduced for a vision-based robotic arm [26]. Scene recognition between virtual and real environments has received little attention. Wallet et al. show via comparison of high and low detailed virtual environments that high detail in the virtual environment leads to better results for tasks such as scene classification and way-finding when applied to real environments [27]. The study was based on the experiences of 64 human subjects.

So far, very little exploration into the possibility of transfer learning between virtual to real environments for the task of environment recognition or scene classification has been performed. Though many of these works are currently preprints and are yet to be published, they already have a high impact, and results are often replicated in related experiments. In terms of scene classification, either LiDAR or photographic image data are considered as a data source for the task, with the best scores often being achieved by deep learning methods such as the Convolutional Neural Network, which features often in state-of-the-art work. Transfer learning features often in these works, either by simply fine-tuning a pre-trained CNN on a large dataset, or training on a dataset and transfer learning weight matrices to a second, more scarce dataset. Inspired by these works, we opt to select photographic data of virtual and real environments before transfer learning by initial weight distribution to a fine-tuned network in order to attempt to use both methods. The successful transfer of knowledge attained in this experiment serves as basis for further exploration into the possibilities of improving environment classification algorithms by considering an activity of pre-training on the infinite possibilities of virtual environments before considering a real-world problem.

III. THE PROPOSED RESEARCH QUESTION AND OUR APPROACH

We propose to answer the research question “Can knowledge be transferred from simulation to real world, to improve effectiveness and efficiency of learning to perform real world tasks, when real world training data are scarce?”. Here, we explain our approach, starting from building the datasets, following with the experiment, choice of models and practical implementation. We include chosen hyperparameters and computational resources in order to promote replicability as well as for future improvement and application to related state-of-the-art problems.

A. Datasets

Initially, two large image datasets are gathered from the following environments:

- Forest
- Field
- Bathroom
- Living Room
- Staircase
- Computer Lab

The first two are natural environments and the final four are artificial environments.

For the simulation data, 1,000 images are collected per environment from the Unity videogame engine via a rotating camera of 22mm focal length (chosen since it is most similar to the human eye [28]) affixed to the viewpoint of a 120cm (3.93ft) robot model, as can be seen in Figure 2. The camera is rotated 5 degrees around the Y axis per photograph, and then rotated around the X axis 15 degrees three times after the full Y rotation has occurred. In total, 6,000 images are collected in order to form a balanced dataset.

For the photographic real-world data, a Google Images web-crawler is set to search and save the first 600 image search results for each environment name. Each set of collected images are sought through manually in order to remove any false results and more data is then collected if needed to retain perfect class balance.

In figure 3 samples of the virtual visual data gathered from the Unity game engine (top row) and photographs of real world environments gathered from Google Images (bottom row) are...
shown. Various similarities can be seen, especially through the colors that occur in nature. Some of the more photo-realistic environments, such as the living room, bare similarity due to the realistic high-poly models for example through the creases in the sofa material. Less realistic environments, such as the bathroom, feature fewer similarities through the shapes of the models, although lighting differs between the two.

**B. Experiment**

With all image data represented as a $128 \times 128 \times 3$ array of RGB values, the datasets are used to train the models. Convolutional Neural Network layers are fine-tuned from the VGG16 network [29] with input layers replaced by the shape of our data, and interpretation layers are removed in order to benchmark a single layer of $2, 4, 8, \ldots, 4096$ neurons. All of these sets of hyperparameters are trained on the simulation images dataset, and an additional set of hyperparameters are then trained on the real images dataset, both for 50 epochs. Following this, all weights trained on the simulation dataset are then transferred to real-world data for a further 10 epochs of training in order to benchmark the possibilities of transfer learning. Thus, both methods of fine-tune and transfer learning are explored. All training of models is via 10-fold cross validation where starting (pre-training) and asymptote (ultimate ability) abilities are measured in order to discern whether knowledge transfer is possible between the domains. A diagram of the experiment can be observed in Figure 4 within which changes in starting ($\Delta S$) and final abilities ($\Delta F$) of the classification of real-world environments are compared with and without weight transfer from a model pre-trained on data gathered from virtual environments.

The goal of the learning process is the minimisation of loss (misclassification) through backpropagation of errors and optimisation of weights. This is possible since all data are labelled, and thus, predictions can be compared to the ground truths. The goal is to reduce the cross-entropy loss [30], [31]:

$$-\sum_{c=1}^{M} y_{o,c} \log(p_{o,c}),$$

where $M$ is the number of classes, in this case, 6, $y$ is a binary indicator of a correct or erroneous prediction, that class $c$ is the true class of the data object $o$, $p$ is the probability that $o$ is predicted to belong to class $c$. If this value is algorithmically minimised, the network is then able to learn from errors and attempt to account for them and improve its classification ability.

The activation layer of the interpretation layer and learning rate optimisation algorithm were arbitrarily chosen as Rectified Linear Units (ReLu) and ADAM. ReLu is defined as $y = \max(0, x)$.

ADAM [32] is a method of optimisation of networks weights during the learning process based on RMSProp [33] and Momentum [34], is generally calculated via the steps:

1) The exponentially weighted average of past gradients, $v_{GW}$ are calculated.
2) The exponentially weighted averages of the squares of past gradient, \( s_{dW} \) are calculated.

3) The bias towards zero in the previous are corrected, resulting in \( v_{dW}^{\text{corrected}} \) and \( s_{dW}^{\text{corrected}} \).

Neural network parameters are then updated via:

\[
v_{dW} = \beta_1 v_{dW} + (1 - \beta_1) \frac{\partial J}{\partial W} \\

s_{dW} = \beta_2 s_{dW} + (1 - \beta_2) \left( \frac{\partial J}{\partial W} \right)^2 \\

v_{dW}^{\text{corrected}} = \frac{v_{dW}}{1 - (\beta_1)^t} \\

s_{dW}^{\text{corrected}} = \frac{s_{dW}}{1 - (\beta_2)^t} \\

W = W - \alpha v_{dW}^{\text{corrected}} \\

\]

where \( \beta_1 \) and \( \beta_2 \) are tunable hyperparameters, \( \frac{\partial J}{\partial W} \) is a cost gradient of the network layer which is being currently tuned, \( W \) is a matrix of weights, \( \alpha \) is the defined learning rate, and \( \varepsilon \) is a small value introduced in order to prevent division by zero.

C. Practical Implementation

In this work, all models were trained on deep neural networks developed in the Keras library with a TensorFlow backend. Implementation was performed in Python. Random weights were generated by an Intel Core i7 CPU which was running at a clock speed of 3.7GHz. RAM used for the initial storage of images was 32GB at a clock speed of 1202MHz (Dual-Channel 16GB) before transfer to the 6GB of VRAM and subsequent learning on a GTX 980Ti GPU via its 2816 CUDA cores.

IV. RESULTS

In this section, the results from the experiments are presented following the method described above. Firstly, the classification ability of the networks trained on virtual data is outlined, then a comparison between networks to classify real-world data initialised with random weight distribution and weights transferred from the networks trained on virtual environments.

A. Initial Training for Virtual Environments

The classification accuracy of the 12 sets of weights corresponding to 2,4096 interpretation neurons respectively to be transferred in the experiment can be observed in Table I. High accuracy is observed with regards to interpretation neurons 8...4096, this is likely due to the CNN generating sets of similar features to the repetitive nature of videogame environments. In order to optimise the rendering of frames to the desired 60fps models, textures and bump maps are often repeated in order to reduce the execution time of the graphical pipeline [35].

<table>
<thead>
<tr>
<th>Interpretation Neurons</th>
<th>Classification Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>33.28</td>
</tr>
<tr>
<td>4</td>
<td>49.69</td>
</tr>
<tr>
<td>8</td>
<td>88</td>
</tr>
<tr>
<td>16</td>
<td>96.04</td>
</tr>
<tr>
<td>32</td>
<td>98.33</td>
</tr>
<tr>
<td>64</td>
<td>98.33</td>
</tr>
<tr>
<td>128</td>
<td>98.16</td>
</tr>
<tr>
<td>256</td>
<td>98.76</td>
</tr>
<tr>
<td>512</td>
<td>97.02</td>
</tr>
<tr>
<td>1024</td>
<td>97.86</td>
</tr>
<tr>
<td>2048</td>
<td>64.08</td>
</tr>
<tr>
<td>4096</td>
<td>93.93</td>
</tr>
</tbody>
</table>

B. Transfer Learning vs Random Weights

The results for the transfer learning experiment can be observed in Table II. The columns \( \Delta S \) and \( \Delta F \) show the change in Starting (epoch 0, no back propagation performed) and Final classification accuracies in terms of transfer versus non-transfer of weights, respectively. Interestingly, regardless of the number of interpretation neurons, successful transfer of knowledge is achieved for pre-training, with the lowest being +3.1% via 2 interpretation neurons. The highest is +48.34% accuracy in the case of 512 hidden interpretation neurons. This shows that knowledge can be transferred as a starting point. The average increase of starting accuracy over all models was +38.3% when transfer learning was performed, as opposed to an average starting accuracy of 16.4% without knowledge transfer. In terms of the final classification accuracy, success is achieved as well, 9 experiments lead to a higher final accuracy whereas are were slightly lower (-0.22% 128 neurons and -3.98% 2048 neurons), and one does not change (32 neurons). The average \( \Delta F \) over all experiments is +7.15% with the highest being +24.56% via 4 interpretation neurons. On average, the final accuracy of all models when transfer learning is performed is 76.34% in comparison to the average final accuracy of 69.16% without transfer of weights.

Overall, the best model for classifying the real-world data is a fine-tuned VGG16 CNN followed by 64 hidden interpretation neurons with initial weights transferred from the network trained on simulation video game environments, this model scores a final classification accuracy of 89.16% highlighted in bold in Table II when both fine-tune and sim-to-real transfer learning are used in conjunction. The majority of results, especially the highest \( \Delta S \), \( \Delta F \), and final accuracy, show that transfer learning is not only a possibility between simulation and real-world data for scene classification, but also promote it as a viable solution in order to both reduce computational resource requirements and lead to higher classification ability overall.
The results serve as strong argument that transfer of knowledge is possible in terms of pre-training of weights from simulated environments. This is evidenced especially through the initial ability of the transfer networks prior to any training for classification of the real environments, but it is also shown through the best ultimate score achieved by a network with initial weights transferred.

V. DISCUSSION

In this section the limitations of this study are discussed and directions for future work to further explore the potential of this method are proposed. From the results observed in this study, there are two main areas of future work which are important to follow. Firstly, we propose to further improve the artificial learning pipeline. Models were trained for 50 epochs for each of the interpretation layers to be benchmarked. In the future the possibility of deeper networks of more than one hidden interpretation layer and also the combinations of the hyperparameters can be explored. The training time of the random weight networks was relatively limited at 50 epochs and even further limited for transfer learning at 10 epochs, although this was by design and due to the computational resources available. Future work could concern deeper interpretation networks as well as increased training time. In this study hyperparameters such as the activation and learning rate optimisation algorithm were arbitrarily chosen, therefore in the future these could be explored in a further combinatorial optimisation experiment. Secondly, simulation to real transfer learning could also be attempted in various fields in order to benchmark the ability of this method for other real-world applications. For example, autonomous cars and drones training in a virtual environment for real-world application. The next step for benchmarking could be to compare the ability of this method to state-of-the-art methods on publicly available datasets, should more computational resources be available, similarly to the related works featured in the literature review [21], [22], [23].

VI. CONCLUSION

In the experiments and results presented in this study, we have shown success in transfer learning from virtual environments to a task taking place in reality. A noticeable set of high abilities were encountered for sole classification of virtual data, as expected, due to the optimisation processes of recycling objects and repeating textures found within videogame environments. Of the 12 networks trained with and without transfer learning, a pattern of knowledge transfer was observed; with all starting accuracies being substantially higher than a random weight distribution, and, most importantly, a best classification ability of 89.16% was achieved when knowledge was initially transferred from virtual environments.

These results provide a strong argument for the application of both fine-tune and transfer learning for autonomous scene classification. The former was achieved through the tuning of VGG16 Convolutional Neural Networks, and the latter was achieved by transferring weights from a network trained on simulation data from videogames and applied to a real-world situation. Transfer learning leads to both the reduction of resource requirements for said problems, and the achievement of a higher classification ability overall when pre-training has occurred on simulated data. As future directions, further improvement of the learning pipeline benchmarked in this study together with exploration on other complex real-world problems faced by autonomous machines are proposed.

VII. ACKNOWLEDGEMENT

This work was partially supported by the Royal Society through the project ”Sim2Real: From Simulation to Real Robotic Application using Deep Reinforcement Learning and Knowledge Transfer” with grant number RGS\[R2\]192498 awarded to D. R. Faria.
REFERENCES


