Transparent Object Classification using 4D CNN

Roland Sireyjol[1,a)] Atsushi Shimada[1,b)] Tsubasa Minematsu[1,c)] Hajime Nagahara[2,d)]
Rin-ichiro Taniguchi[1,e)]

Abstract: Despite the increasing popularity of image processing techniques for object classification, no efficient technique has been found to classify transparent objects. In this paper, we tackle this issue by proposing various uses of Convolutionnal Neural Networks, and comparing their efficiency. The method considered as the most efficient will then be more attentively studied, in order to identify its main features.

Keywords: classification, light field, neural network, distortion

1. Introduction

In recent years, image processing studies improved a lot, wether through the developpement and exploitation of hand-made features, or by the use of deep learning, and especially neural networks [1]. These improvements are essential for computer vision, and most of us use softwares that need this kind of image processing techniques. Object identification or facial recognition, for example, are widely used by very popular sotfwares. Since transparent objects like glasses or bottles are everywhere around us, identifying them is as critical as identifying any other object. However, despite the increasing amount of object classification methods, no efficient and easy to use techniques have been found yet for transparent object classification (TOC)[1], [2], [3]: Despite showing better results than usual object classification techniques, hand-made feature detection using light field distortion (LFD) [4], [5] is tedious to use and shows, in the best conditions, an 85 % accuracy. In a previous paper [6], we tackle this issue by using a 3 dimension CNN with a light field dataset. Nevertheless, this paper presented only one kind of 3 dimension CNN, and did not offer a better accuracy than the hand-made feature. In this paper, we propose multiple approaches to tackle this challenge using various CNN architectures and combinations. Comparing those approaches' pros and cons regarding their ressource consumtion and final results can help us identify which method is the most efficient, and why. With an easy to use and efficient deep learning classification system, we can study in detail its features and realize efficient TOC.

2. Analytics methodology

Our dataset is provided by a light field camera, and has four dimensions: (s,t,u,v) [4]. However, some of the CNN studied here use 3 dimensions only. This section will therefore present the dataset, along with the method we used to adapt the 4D dataset to 3D CNN. Ensues a section explaining all of the caracteristics that we use to compare our different approaches of TOC.

2.1 Light field dataset

The data used for this study was obtained with a ProFUSION-25C [7] camera, capturing 5*5 VGA images simultaneously from 25 different viewpoints. Each image is originally 640*480 pixels, but they have been cropped to 480*432 pixels, and set in black and white.

Data obtained from a LFC extends on four dimensions (s,t,u,v): the viewpoint plane (s,t) can be associated to the position of the camera among the 5x5 ProFusion25 cameras, and the image plane (u,v) can be associated to the usual width and height coordinates of a pixel for each image captured by the ProFusion-25C cameras(cf Fig. 1). Our dataset contains 20 different objects that were captured in front of 10 backgrounds. In order to produce a coherent dataset for CNNs, we split the data between a training and a validation set, regarding the background repartition: data obtained from 8 backgrounds are given to the training set, and data from the 2 last backgrounds are used for the validation set.

One of the difficulty for TOC is the size of original images: unlike usual object classification, the entire image changes with the background. For this reason, when greatly reduced images can be used for usual objects (28*28 for MNIST dataset), it cannot be used for TOC, especially for the most complex features (for example, LFD feature [4] could not be detected).

2.2 Light field data adaptation to CNN

4D CNN can use the dataset as it is, however it needs to be adapted when using 3D CNNs: Following the same methodology than our previous paper [6], viewpoints (u,v) from a single direction of the (s,t) plan has been used, representing five consecutive viewpoints. If this direction is horizontal , viewpoints from the

[1] Kyushu University, Japan
[2] Osaka University, Japan
[a)] roland@limu.ait.kyushu-u.ac.jp
[b)] atsushi@limu.ait.kyushu-u.ac.jp
[c)] minematsu@limu.ait.kyushu-u.ac.jp
[d)] nagahara@ids.osaka-u.ac.jp
[e)] rin@kyudai.jp

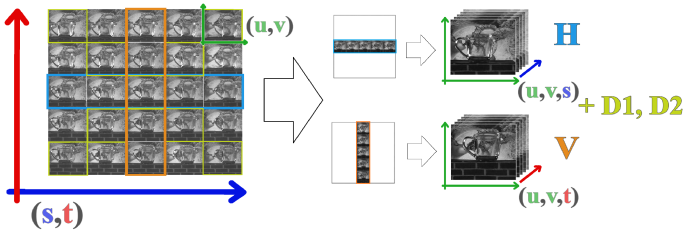

Fig. 1 Selection of viewpoints among LF data of dimension (s,t,u,v). Horizontal direction (H) in orange, vertical direction (V) in green, and diagonals (D1,D2) in blue.

3rd line of the (s,t) plan are used as shown in Fig. 1; If direction is vertical, we use viewpoints from the 3rd column.

In the study on LFD hand-made features [4] , the best combination of viewpoints for classification are using those on the central raw and column of the (s,t) plan, along with the two main diagonals. For this reason, we will use the same four axis in our studies. Moreover, in order to compare them, each CNN of those directions are using the same architecture and initial parameters. Those directions will be referred as H (horizontal), V (vertical), and D1 D2 for the diagonals. This could also allow us to realize a complete transfer learning from one direction's CNN to another.

2.3 Analytics strategy

For every different use of CNN, specific parameters will be measured to compare their efficiency: their resoucre consumtion and the results it produces.

- Resource consumtion: Our goal is to propose a practical, cheap, fast method to classify transparent objects. We therefore consider RAM, memory storage, and processing time: if these values are too high, the process is not a viable option. A particular note will be added regarding the training process, since we only need to do it once to exploit our CNN.
- Final result: Once the data is processed, we consider the accuracy of our system, with the objective to obtain a higher accuracy than the hand made feature. However, for some systems, the accuracy drastically decreases when inital parameters are slightly different from ideal values. We therefore consider this characteristic as the stability/robustness of our system.

3. Different CNN approaches

Light field dataset offers a vast possibility of approaches to process its data, that has different computationnal cost, accuracy and stability. Here are the different approaches that we studied in this paper, along with their characteristics.

- 3D CNN from all directions: For each of the four directions (H, V, D1, D2), our 3 dimension CNNs use the same architecture presented if Fig. 2, along with the same initial values for training (learning rate, random initial weights and biases...). 3D CNNs are the easiest and lightest studied option, however accuracy from cross validation still remains low and unstable.
- 3D CNN Combination: From those four 3D CNNs, we extract the last layer's output, associated to the probability for each element of the batch to belong to one of the classification category. Considering the estimations made by theses four CNN allows us to overcome the error of one CNN with the output of the 3 others. It is more complex to realize than simple 3D CNN, but gives way better results with the highest stability. Illustration of 3D Combination is given in Fig. 3

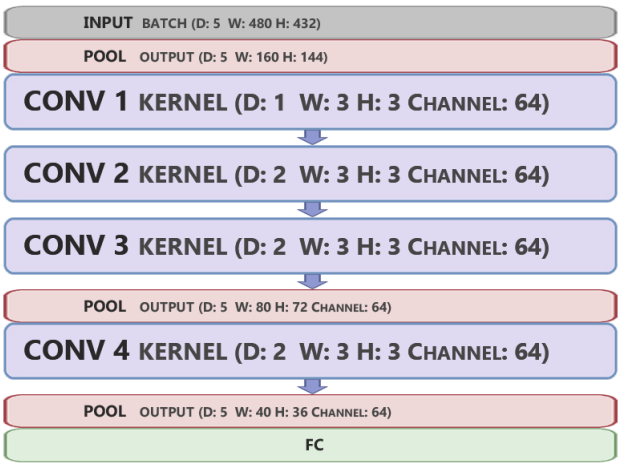

Fig. 2 3D CNN architecture.

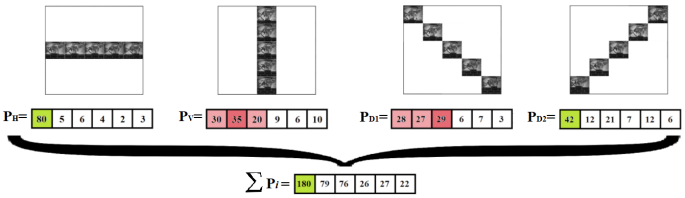

Fig. 3 3D CNN Combination, using the output of all four directions to produce its result.

- 3D transfer learning: Since CNNs for all 4 directions use the same architercture, we can train one direction with the trained values of another direction as initial values (For example, retrain the final V CNN for the horizontal direction: V is the original CNN, and H is the transferred CNN.). Transfer learning is easier to realize than 3D Combination and gives similar accuracy, but does not offer the same stability.

  Additionally, various 3D combination can also be done with 3D CNNs from transfer learning. We combine transfer learning CNN having the same original direction, therefore four combination can be produced.

  Multiple transfer learning has also been studied, by training a direction from another transfer learning CNN values. However, the increase of accuracy were not proven convincing (about 1.5% increased accuracy, for a higher complexity).
- Hybrid approach (3D to 4D CNN): Using the four 3D CNN, we extract the ouput of the third pooling layer (just after the last convolutionnal layer) recompose it as a 4D data following the (s,t,u,v) correspondences, before using it as the input of a 4D CNN. The complexity of this system rapidly revealed itself too ressource-consuming, and its accuracy were found relatively low. For these reasons, a quantitative study of this approach were considered enough to disband this option.

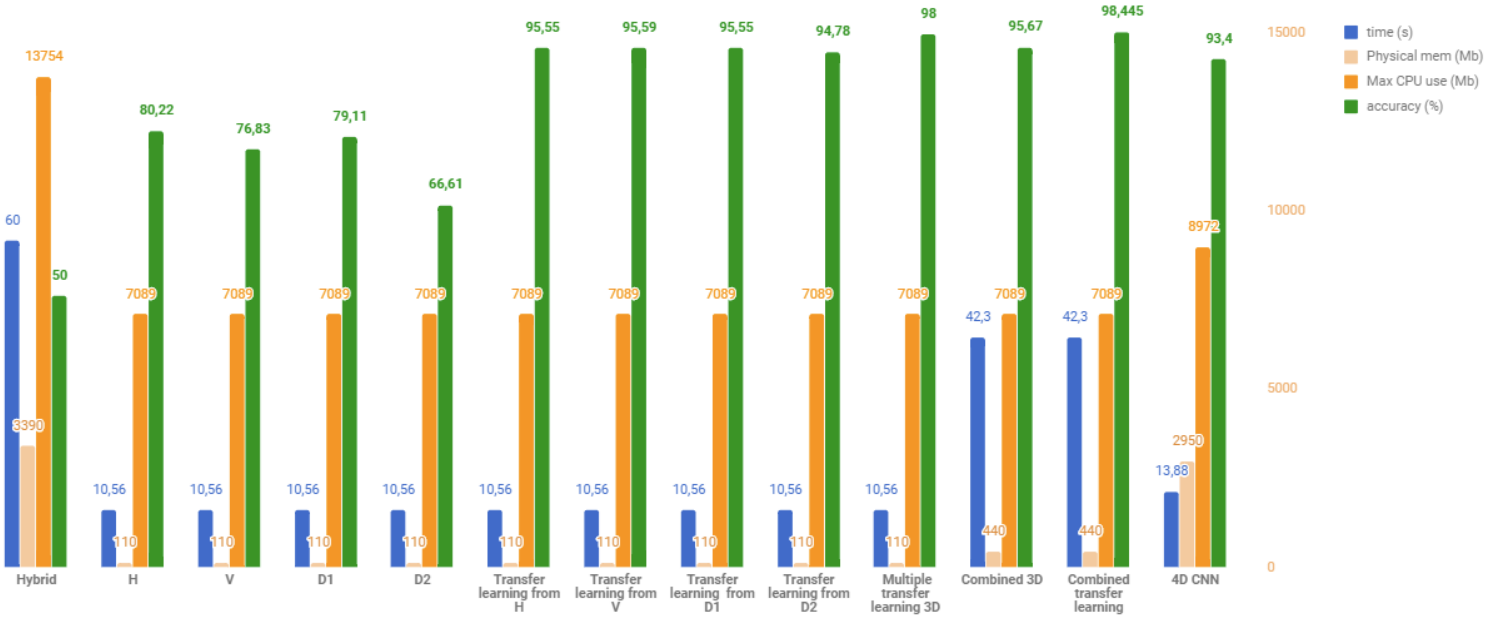

Fig. 4 Results for all approaches.

- 4D CNN: Unlike 3D CNN, 4 dimension CNN uses the entire data obtained from a light field camera, making its implementation way easier than any other approach. Moreover, the standard architecture of our 4D CNN is extremely simple, with only two convolutional layer and one fully connected layer. While it still consumes more resources than a single 3D CNN, it gives comparable results than 3D combination process, way faster. However, its memory consumption (especially RAM) is higher than other methods, and changing this CNN (by adding a convolutionnal layer, for example) makes it even worse.

By studying different approaches to classify transparent object features, we can select the most efficient technique, and dive into the features it has learnt to in order to produce a cheaper, more efficient TOC technique. Optimization is also a key to improve this TOC technique.

4. Results and interpretation

The entire set of results is held in Fig. 4.

4.1 General observation for resources consumption

Memory storage is, for every approach, almost entierly consumed by the saved model of our CNN. For this reason, all 3D CNN approaches have the same memory storage consumption. 3D CNN Combination needs models for each direction, and therefore costs four times as much as a single 3D CNN approach. 4D CNN's saved model is, without surprises, way bigger than 3D approaches.

The maximum use of RAM is also mostly done by the CNN process, and each 3D CNN cost as much as the other. Moreover, since combining 3D CNN is made by obtaining probabilities of each CNN sucessively, the maximum RAM consumption is almost the same as a single 3D CNN. 4D CNN, however, is unsurprisingly higher.

Processing time is also the same for each 3D CNN, and four times as much for combined 3D CNN.

The actual values of those parameters are not so important, since it can change on other machines and codes: comparing them together is what is important here.

4.2 Interpretation

As described earlier, the less efficient technique is, by far, the 3D to 4D CNN approach. Immediatly after comes single 3D CNNs.

Since each 3D CNN is using the same initial parameters, each CNN is not using the best values for their specific direction (H, V, D1 or D2). It should not change the final result by much, however, because of our relatively small dataset, 3D CNN are not very robust, which is why cross validation accuracies are so low (average 73.66%). However, its resource consumtion is the lowest possibility in every domain.

When looking at the obtained results, we observed that, for single 3D CNN, estimated class for an input was often a tie, whith two or three classes having almost the same probability. When combining 3D CNN together, more data could be considered for the same object from different processing techniques, which in the end corrected those errors. 3D Combination therfore has an accuracy of 95.67%, and is the most robust technique. However its processing time is four times higher than a simple 3D CNN.

Transfer learning process also shows great results, with an accuracy around 95 %, however its robustness is still a huge problem, being sensively the same than usual 3D CNNs. 3D CNN Combination from transfer learning is the most accurate system, however its improvements comparing to normal 3D Combination is still relatively small: With the same resources consumtion results as standard 3D combination, its higher accuracy is counterbalanced by its increased complexity and training process. Combination using transfer learning is one of the best option, with one main flow: its processing time. Using only 3 out of 4 directions could save 25 % of the processing time, with little impact on ac-

Fig. 5 Image from the background 7: red, orange, yellow, the shapes of different transparent objects. In blue, characterisit shapes of this background: the similarity between them and object shapes explain the loss of accuracy faced by our CNN.

curacy. Further improvements could be made, but its processing time would still be important.

5. 4D CNN: the most efficient option

Two major approaches can be considered as the best options for TOC: 4D CNN, and Combination of 3D CNN. However, 3D CNN combination is way more complex than 4D CNN, and its processing time can hardly be reduced to the same level than 4D CNN. Moreover, the main reason our 4D CNN has a lower accuracy is due to the size of our dataset (which can be improved when applying this approach in another enivronment than research), and one specific element that we will explain.

5.1 Comparison with 3D CNN Combination

5.1.1 Resources consumption

The complexity of 3D CNN combination makes it harder to modify: our previous study [6] shown that the currently used architecture is the most efficient, therefore modifying it would only damage the current results. Moreover, the 3D CNN combination is 3.2 times slower than 4D CNN, and can hardly be improved.

5.1.2 Final results

3D comparison approach gives better accuracy than 4D CNN (95.67% against 94%), because of its stability: 4D CNN is extremely small, and with such a small dataset, it is even more sensitive to changes in its initial values, validation set, and number of training steps, even getting a 60 % accuracy with a specific validation set. As explained before, our validation set contains all images captured from two different backgrounds. Cross validation process revealed that, when a specific background was contained in the validation set, accuracy would drop drastically, unlike 3D CNN combination (for which accuracy drops less than 5 %). Unlike other backgrounds, this one display shapes very similar to transparent object edges: LRP analysis and our previous study showed that 3D CNN were sensitive to the edge of the transparent object. Since 4D CNN shows the same tendancies than our 3D CNN (accuracy drops and increases for the same validation sets), it also seems to grasp this kind of features too. Fig. 5 shows an image of this background, illustrating this idea.

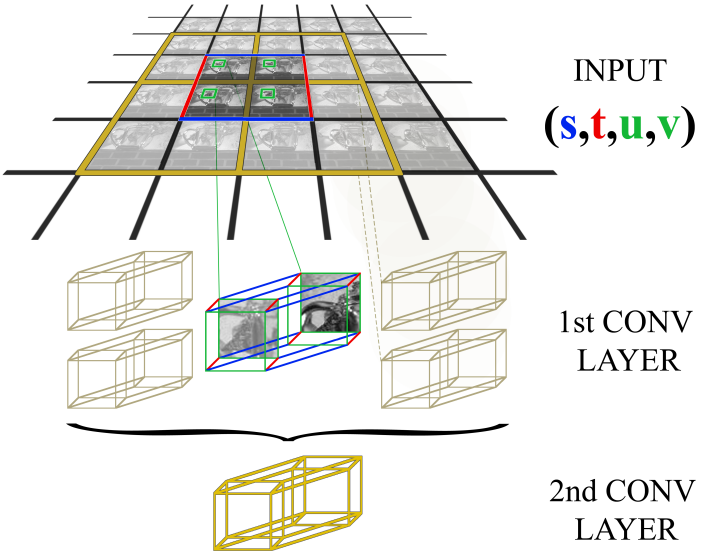

Fig. 6 4D Convolution explanation. The output contained in the yellow hypercube (bottom) uses data from all viewpoint located in the yellow square (top).

If processing time is not a key point of the classification process, 3D Combination is worth considering, especially when it is combined with transfer learning.

5.2 4D convolution: a tool for LFD detection

Since our dataset is produced from a LF camera, 3D convolution could compare a specific area in the (u,v) plane from the same area, along the third axis. For every convolutionnal layer, we could compute datas processed from the surrounding viewpoints: For example, at the first conv layer using the third axis (”depth”), viewpoint N could be compared to viewpoints N and N-1. On the second convolutionnal layer, outputs from viewpoints N-2, N-1, N could be compared with output from viewpoints N-1, N, N+1. This way, features like distortion could be eventually identified by the CNN, however, considering only one axis greatly limitated the range of viewpoints that we can compare.

4D CNN extends this property even further, comparing a viewpoint with its surrounding viewpoints in the (s,t) plan, and not only one axis. For this reason, as illustrated in Fig. 6, even with only two convolutionnal layers, the number of viewpoints considered to produce the output of a (s_1, t_1, u_1, v_1) point is more important.

Considering more viewpoints can greatly help identifying Transparent object features, and especially distortion.

5.3 4D CNN: Evolution and optimization

As seen earlier, 4D CNN is resource consuming on every aspect, and improving its accuracy is as important as reducing its memory consumtion and processing time. Various ways to optimize our 4D CNN can be established, either by limitating the original input, or by deleting some channels.

5.3.1 Reduce the input

In the default situation, we transmit to the CNN all 25 viewpoints obtained from the ProFusion-25C camera. Since the differences between objects is mostly caused by distortion of the back-

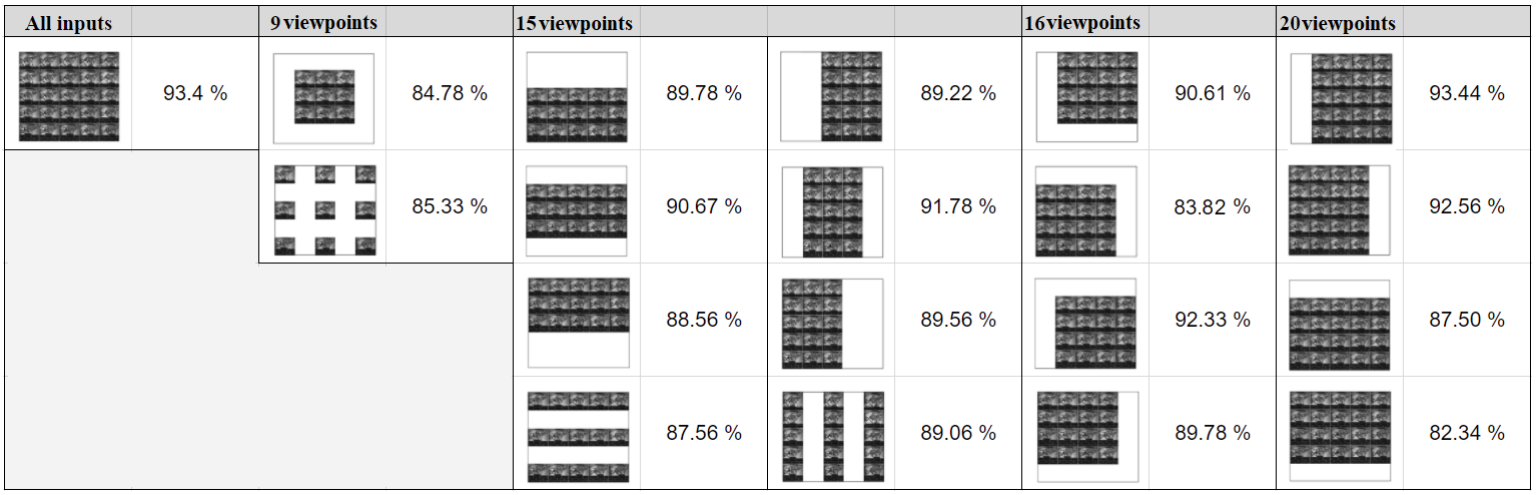

Fig. 7 Various possibilities of reduced input, along with their accuracies (percentage). Other kinds of input have been tested, with the same structure.Average accuracy with 9 viewpoints: 85,055%. 15 viewpoints: 89,523%. 16 viewpoints: 89,135 %. 20 viewpoints: 88,96 %. With smaller amount of viewpoints, accuracy is around: 6 viewpoints: 77.29% ,4 viewpoints: 75.18 % , 3 viewpoints: 70.61 % , 2 viewpoints: 67.818 % , 1 viewpoint: 63.67 % .

ground, which is contained in a small part of the image, reducing the size of each image in the (u,v) plan cannot be overused. However, reducing the number of viewpoints transmitted can help reducing the computation cost without damagin the accuracy. Fig. 7 presents different options that were studied, along with their accuracy.

Reducing the data along the (s,t) plan seems to be a powerful option, but its impact on the accuracy and stability is important. However, considering those options, interesting results are obtained:

Interestingly, unlike we could expect, accuracy with 15 viewpoints is higher than accuracy with 16 or 20 viewpoints: this can be explained by the high standard deviation of 16 and 20 viewpoints' possibilities. This high standard deviation is caused by the fact that, when using 16 viewpoints (4*4 viewpoints on the (s,t) plan), one of the four possibility only has 76.94 % cross-validation accuracy, and when using 20 viewpoints, one possibility only has 82.34 %.

Some specific images (or combination of images) are extremely useful for classification, when others are not. Since we cannot predict which viewpoint will be useful or not for a new dataset, such method has limited results on optimization.

Despite those surprising results, 4D CNN stays the best option when using the full dataset.

5.3.2 Delete some channels

Even though our CNN is already small, deleting some channels might reveal itself a viable option: by considering the impact of each channel in a similar way than our previous paper [6], we can identify the least useful channels, and delete it completely. If It doesnot impact the final accuracy, the CNN would have been optimized efficiently.

However, results showed that, for our 4D CNN, every channel was important, and even deleting the least efficient channel would still impact the final accuracy. This method is not viable for 4D CNN, but using it on 3D CNN Combination actually helped reducing its computationnal cost (Nevertheless, this optimization was not enought to make it competitive).

5.3.3 Change the architecture of the CNN

The standrad architecture only uses 2 convolutionnal layers and a fully connected layer: it is not deep enough to grasp complex features, and mainly focuses on the shape of objects it can identify (along with relexion of light ...). A deeper CNN could eventually learn to identify features like distortion. However, adding a third convolutionnal layer between a pooling layer indtroduced a strong overfitting, that could not be corrected through regularization. Moreover, adding a single layer greatly increase the resource consumption of our system, and the processing time is almost three times as important as two convolutionnal layers.

Increasing the dataset with new LF images submitted to different background and illumination could allow us to use a deeper CNN, in order to learn more complex features like distortion. Using deep learning techniques and a LF dataset, we could eventually extend this study to develop a 3D map of the distortion caused by transparent object refraction.

5.3.4 Improving the dataset

As presented earlier, we have a rather small dataset, which causes our CNN to overfit rapidly before learning complex features (which are exacly what we need in this study). As a result, the architecture of our CNN must stay simple, and classification is highly unstable. Fig. 8 illustrates this instability in the results.

However, with a bigger dataset, overfitting can be prevented, and accuracy can increase. All previous methods to optimize the CNN can then be used.

5.4 Conclusion

Since transparent object are everywhere around us, developping efficient techniques to identify them is very important in computer vision and robotics. However, usual object identification techniques do not work with transparent object, and new approaches must be used to tackle this problem. This paper proposes different approaches in neural networking for transparent object classification using a light field dataset, and identifies two

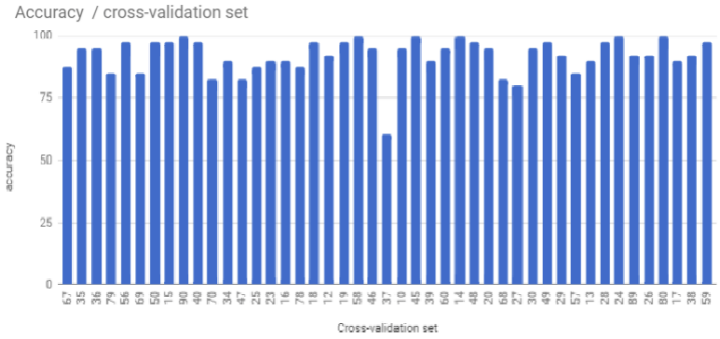

Fig. 8 Accuracy for each validation set. Each element of the horizontal axis represents a new distribution of images between the validation and the training set. As explained before, our dataset is made of 200 LF images (20 objects, 10 backgrounds). If images from background 6 and 7 are kept in the validation set, on the horizontal axis will appear "67". Standard deviation is 7.4 %.

major techniques: 3D CNN Combination, which uses the output of four 3D CNN, and 4D CNN. Because of its result and potential, 4D CNN are identified as the best option for TOC, and a deeeper study of this option is made in order to optimize it, whether by changing its architecture, deleting the less useful channels or reducing the input size. In this case, none of those methods were viable option, since 4D CNN need a much bigger dataset to grasp good features.

References

[1] M. Fritz, M.J. Black, G.R. Bradski, S. Karayev, T. Darrell, An additive latent feature model for transparent object recognition, in: Neural Information Processing Systems (NIPS), 2009

[2] D. Miyazaki, K. Ikeuchi, Inverse polarization raytracing: estimating surface shapes of transparent objects, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005, pp. 910917.

[3] G.S. Settles, Important developments in schlieren and shadowgraph visualization during the last decade, in: International Symposium on Flow Visualization (ISFV),2010.

[4] Y. Xu, K. Maeno, H. Nagahara, A. Shimada, R. Taniguchi, "Light field distortion feature for transparent object classification, " Kyushu Univ. Fukuoka, February 2015

[5] K. Maeno, H. Nagahara, A. Shimada, R. Taniguchi, Light field distortion feature for transparent object recognition, in: IEEE Conference on Computer Vision and Pattern

[6] R. Sireyjol, A. Shimada, T. Minematsu, H. Nagahara, R. Taniguchi, "How does CNN grasp transparent object features? ", Kyushu Univ. Fukuoka, January 2018

[7] https://www.ptgrey.com/Content/Images/uploaded/KB-Data/ProFUSION_ 25_ datasheet.pdf.Recognition (CVPR), 2013, pp. 27862793.