*Signal Processing in the Context of Neurotechnologies 2021*

The Müller-Lyer illusion in CNN trained for 3D object height estimation

Anton N. Mamaev, Ivan A. Gorbunov

Saint-Petersburg University

Keywords: Convolutional neural networks, Optical illusions, Cognitive modelling, Image processing, Regression problem

# Abstract

Convolutional neural networks are structurally inspired by the visual cortex and therefore provide an opportunity for modelling processes underlying human perception. However, the research on the effects of optical illusions in such models is currently insufficient and lacks plausible simulations that could explain their causes.

Previous attempts to investigate the Müller-Lyer illusion using CNNs were limited to binary classification and 2D pictures [1]. We propose a more natural approach that returns quantitative results suitable for statistical analysis and comparison with human performance.

To recreate real-life object size measurement, we fit a feed-forward convolutional model to estimate the height of a cuboid in a presented picture. To incorporate perspective effects perceived in the physical world and enhance model validity, we use renderings of a 3D scene as training images and the cuboid's randomized height parameters as target values. Each picture is an inside or an outside view of the cuboid's edge from a randomized distance. The geometry of this scene resembles the interiors and exteriors of various buildings.

The model processes 200x200 grayscale images and outputs the estimated height with a single linear-activated output neuron, solving a regression problem. By adjusting parameters and minimizing the mean squared error between estimated and target values, the model maximizes the accuracy of estimations.

Using transfer learning, we acquired height estimations for the inward and outward-facing arrow images of the Müller-Lyer optical illusion and found that the inward-facing arrows are consistently estimated larger than the outward-facing ones the same way it happens in human perception.

This evidence reinforces the idea of a connection between the Müller-Lyer illusion and the effects of perspective found in three-dimensional environments and encourages further research of illusions and perception using convolutional neural networks and other quantitative methods.

# Introduction

The Müller-Lyer illusion is a classic optical illusion in human perception. However, the causes and underlying principles of the illusions are debated [2]. While some researchers attribute the phenomenon to depth cues, size constancy and spatial perception others explain it with weighted mean and receptive fields activation summation. Furthermore, the research mostly covers the observation of natural phenomena while artificial modelling and simulation of the process are not being conducted. We believe that comprehensive research in this new direction could be valuable for illusions research and propose convolutional neural networks as means to model various phenomena in human perception.

There is at least one published research paper in the scope of optical illusions computational modelling that covers the Müller-Lyer illusion [1]. However, that study also has several problems we attempt to solve in the current research.

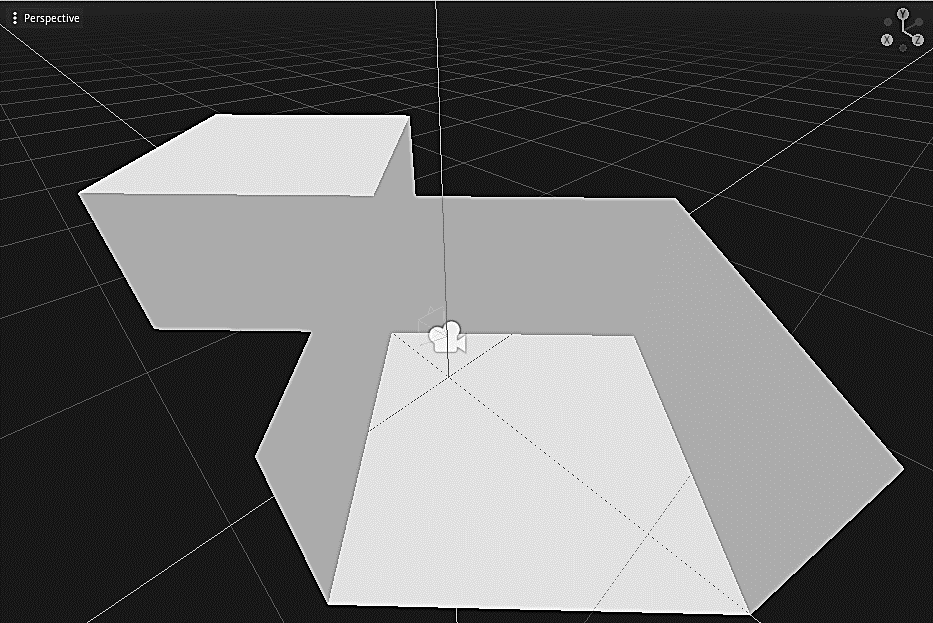
Firstly, there was no spatial perception involved. All the stimuli were plain 2D lines and arrows, making it impossible to test if depth cues could be involved in illusion occurrence. Secondly, the neural network architecture used in the study was HMAX, a complicated convolutional neural network with variable filter resolutions and orientations made to represent the visual cortex as closely as possible. While its' strong resemblance to neural structures of the human brain is valuable, usage of a highly-specific and complex architecture makes it hard to both replicate the results and investigate the inner states of the model. Finally, the learning problem in the research was binary classification, enabling only two possible outcomes — the right one and the wrong one. That alone undermines the experimental results as the chance of a lucky guess is a vast 50%. Moreover, output data of classification problems, in general, lack the continuity necessary for post-hoc quantitative analysis.

In our study, we revisit convolutional models in the Müller-Lyer illusion research to find evidence whether the illusion is a consequence of spatial perception development as proposed by one of the interpretations. We have to assume that convolutional neural networks share enough similarities with the visual cortex to be a representative model at least for simple visual phenomena. We also have to state that a neural network is a ‘blank slate’ before any training and the outputs of a trained network depend on the training data and learning problem. Now, as perspective explanation attributes the illusion to the measurement of volumetric bodies, our approach to prove or disprove that was to train a height measurement neural network on a series of 3D imagery and then present it the Müller-Lyer illusion. In case the model estimations are consistently biased in accordance with original studies in humans, we have to further consider the possibility of depth perception development being a major cause of the Müller-Lyer illusion.

# Methods

## Dataset

While an initial research idea was to fit the model with indoor and outdoor photos of buildings with variable height and position, it was found to be ineffective as it is hard to determine the exact height of a building and the number of photos has to be tremendous. Instead, we came up with a dataset generator — an auxiliary application that renders series of images from a 3D environment built with Godot 3.3.3 open-source game engine. The 3D scene shown in Figure 1 consists of two cuboid meshes and a virtual camera (a) facing the mutual edge (b) of the meshes while being inside one of them. At any given moment one of the meshes is hidden so the picture taken by the camera is either the interior or exterior view.



a

b

Figure 1. 3D environment

To prevent overfitting and to ensure that the model performs well regardless of object and camera positions we introduced additional spatial parameters other than height. In total, there were four variables of the scene that are randomized in the given ranges for each picture taken:

1. Mesh height, also mutual edge length (2,4.2) — target value
2. Camera distance (0,5)
3. Object angle (-10,10)
4. Camera angle (-25,25)

The randomization ranges were limited to ensure that the mutual edge is always in the camera’s field of view and it is possible to estimate its’ height.

All of the output images as shown in Figure 2 were limited to grayscale and 200×200px. resolution to enhance training speed. The filenames included the target value rounded up to sixth number after the decimal point exported directly from 3D scene height variable. We also created four special stimuli similar to the original Müller-Lyer illusion exclusively for testing purposes. They were strictly held out of training dataset to ensure their estimations are based on transfer learning only.

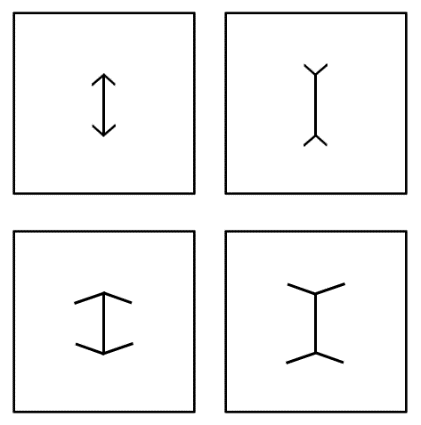
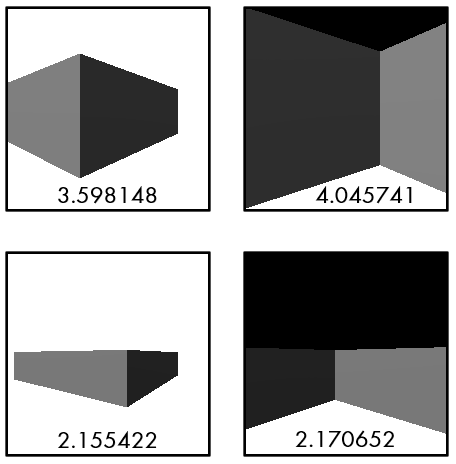


Figure 2. Input data examples

Overall, the final training dataset comprised 500 computer-generated images and a batch of 30 other images was held out to test estimation accuracy.

## Neural network architecture

We used Keras [3] framework to compile and fit a feed-forward network consisting of ten layers in total:

1. Convolutional layer 1 (200×200×1 input, 32 3×3 filters)
2. Max Pooling layer 1 (2×2)
3. Convolutional layer 2 (64 3×3 filters)
4. Max Pooling layer 2 (2×2)
5. Convolutional layer 3 (128 3×3 filters)
6. Max pooling layer 3 (2×2)
7. Flatten (reshape 3D array to 1D vector)
8. Fully-connected layer 1 (1024 neurons)
9. Fully-connected layer 2 (512 neurons)
10. Fully-connected layer 3 (1 neuron)

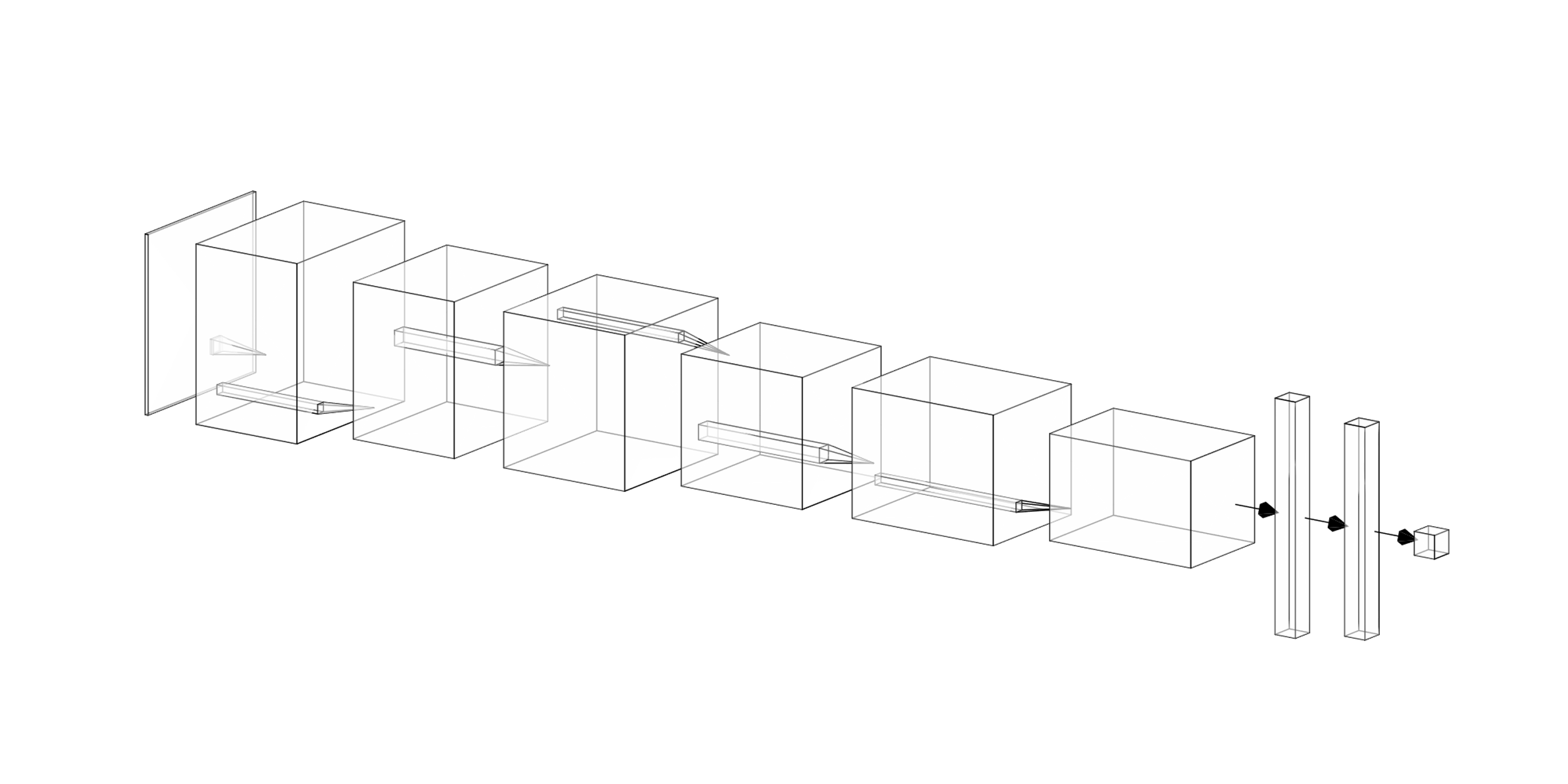
A visual overview of the architecture is given in Figure 3. Array dimensions are scaled to fit and should not be referenced as spatially correct.

Figure 3. Neural network overview

The input layer has the exact shape of a 200×200 pixels grayscale image while the output converges to a single neuron. Most of the layers have ReLU as an activation function with an exception of the output neuron activated lineally. The latter enables the model to provide both positive and negative numeric estimations without being limited by maximum or minimum values.

As the learning problem is regression and the output is a single numeric value, the loss function is Mean Squared Error, as opposed to Cross-Entropy Error for classification problems.

## Procedure

The overview of the preparatory and main stages of model training is shown in Figure 4.

First of all, the dataset generator synthesizes a designated number of images necessary for training, 500 for our final models. The spatial parameters get randomized, the picture gets taken and saved in a folder and the process loops until the set number of images is reached. All of the images are imported by a pre-processing Python script, get shuffled, the target value is extracted from the file's data and the image data is converted to a 200×200×1 matrix, both get rescaled and appended to the Y and X array respectively. The Y array is composed of standardized height values, while the X array is composed of matrices with pixel values rescaled from (0,255) to (0,1).

The model then gets compiled and fitted for 30 epochs with a learning rate of 0.001. Items of array X get fed through the network and the output value of a single neuron in the last layer is compared with the target value from array Y. The loss is calculated with the mean squared error function and changes are made to the parameters of the model depending on the extent of the discrepancy through backpropagation. Around 30 epochs the loss gets as low as <0.001.

In this study, we had two separate testing phases. The first one tests the accuracy of estimations on a novel dataset unknown to the model while the second one tests whether the model can transfer its' training to a completely different kind of stimuli — the arrows of the Müller-Lyer illusion. The second phase begins only if the model performs with high accuracy at the first phase.

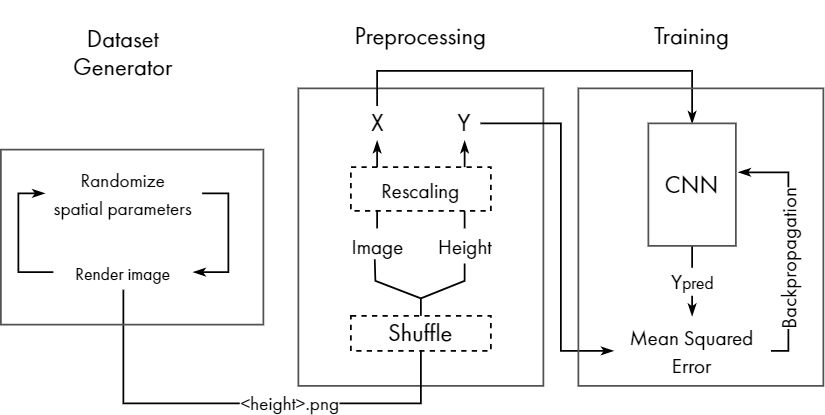


Figure 4. Data pipeline

# Results

## Height estimation accuracy

The fitted model's height estimations for 30 random images not included in the training dataset were compared to the actual height values. The results illustrated in Figure 5 show high estimation accuracy with a mean deviation of 0.053114 and a correlation coefficient of 0.998854. Raw data is presented in Table 1.

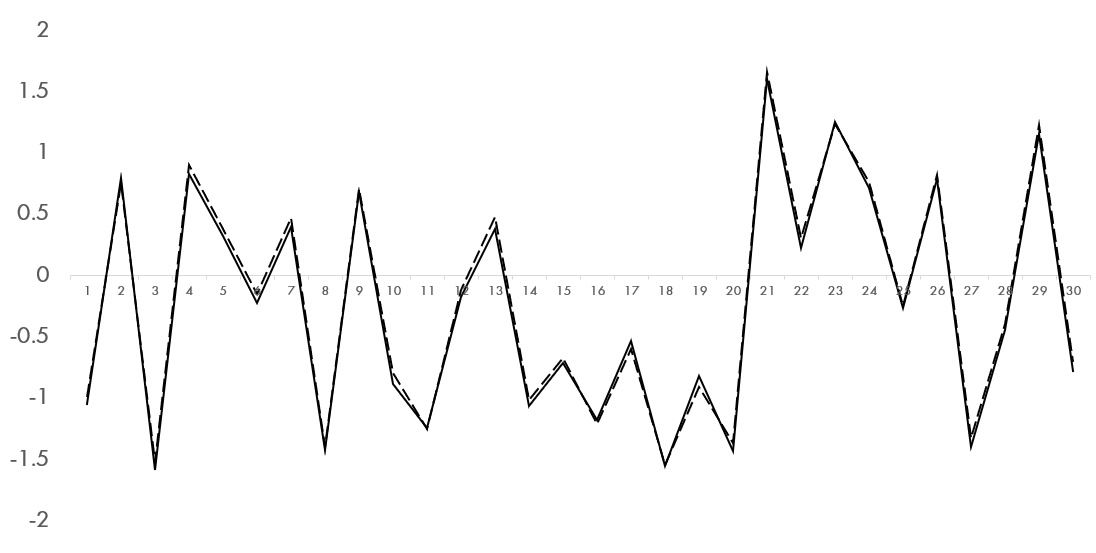


Figure 5. Estimation accuracy

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| -0.99725 | 0.75132 | -1.52477 | 0.896639 | 0.380402 | -0.15586 | 0.462907 | -1.40121 | 0.704117 | -0.79454 | -1.25422 | -0.12185 | 0.478146 | -1.01683 | -0.67781 | -1.21133 | -0.60048 | -1.5506 | -0.90883 | -1.37262 | 1.655873 | 0.305259 | 1.234963 | 0.756578 | -0.24821 | 0.819196 | -1.3303 | -0.3997 | 1.223734 | -0.7062 |
| -1.05337 | 0.794811 | -1.58703 | 0.821801 | 0.320473 | -0.2264 | 0.392117 | -1.42639 | 0.675531 | -0.8898 | -1.24876 | -0.17984 | 0.374568 | -1.06528 | -0.72024 | -1.18067 | -0.54084 | -1.55434 | -0.82615 | -1.43319 | 1.597133 | 0.224779 | 1.2505 | 0.7133 | -0.27217 | 0.784747 | -1.39949 | -0.44908 | 1.154376 | -0.78613 |

Table 1. Accuracy raw data

## The Müller-Lyer illusion effect

To verify that the Müller-Lyer illusion consistently affects estimations we fitted a total of 10 models with the same training dataset and recorded the outputs for two pairs of stimuli made to cause the Müller-Lyer optical illusion. As it is shown in Figure 6, all of the models have estimated the lines with inward-facing arrowheads to be slightly longer than with the outward-facing ones. The mean deviations for classic and wide stimuli pairs are 0.130740063 and 0.274046172 respectively. The deviation is higher by 0.143306109 in the second pair of stimuli with wider arrowheads that resemble an inside and outside edge of a 3D object used in training. Raw data is presented in Table 2.

|  |  |
| --- | --- |
|  |  |

Figure 6. Illusion stimuli deviation

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Classic stimuli | | | | | | | | | |
| 0.072521 | -0.018522 | 0.072521 | -0.018522 | 0.072521 | -0.018522 | 0.072521 | -0.018522 | 0.072521 | -0.018522 |
| 0.131267 | 0.027986 | 0.131267 | 0.027986 | 0.131267 | 0.027986 | 0.131267 | 0.027986 | 0.131267 | 0.027986 |
| Wide stimuli | | | | | | | | | |
| 0.249548 | -0.02488795 | 0.249548 | -0.02488795 | 0.249548 | -0.02488795 | 0.249548 | -0.02488795 | 0.249548 | -0.02488795 |
| 0.326664 | 0.08317863 | 0.326664 | 0.08317863 | 0.326664 | 0.08317863 | 0.326664 | 0.08317863 | 0.326664 | 0.08317863 |

Table . Illusion stimuli raw data

# Discussion

In the current research, we show that the Müller-Lyer illusion causes a convolutional neural network to provide biased length estimations. The effect is comparable to the effect in human perception. Considering that the model was fitted only to estimate the height of 3D objects, we attribute the occurrence of the Müller-Lyer illusion to it being caused by the particularity of spatial perception itself.

As the Müller-Lyer illusion consistently affects height estimation in convolutional models fitted with spatial imagery, the architecture of a neural network developed for length and height measurement in spatial perception is sufficient to cause the illusion in its classic form. The model fitted with the sole purpose of estimating the height of 3D is essentially a model of a visual subsystem in development, and the fact it is affected by stimuli such as the Müller-Lyer illusion bears evidence of the visual illusions and functions of the vision itself being linked.

While this suggests a strong connection between illusion and spatial perception, the findings do not necessarily conflict with explanations other than the one attributing the illusion to spatial perception. Though spatial vision is viewed as the essential cause of the phenomenon, the weighted mean and neural activation summation could probably be the exact neural processes that provide a foundation for spatial perception itself.

# Conclusion

Our research has reached two main conclusions. First, it shows that the Müller-Lyer illusion causes a persistent length estimation bias in neural networks largely similar to the effect it causes in humans. As the effect was found in a model fitted from scratch solely for the task of 3D object height estimations, we suspect the illusion to be caused directly by the development of spatial perception.

Second, we have found out convolutional neural networks are capable of accurate estimations for complex spatial scenes and viable to use in modelling and simulation of human sensory and perceptive processes.

Further investigation is required to find out whether the effect exists for other variations of the illusion, compare the model's estimations with human performance and determine the internal parameters and processes of the neural network that contribute to the visual phenomena.

# References

[1] A. Zeman, O. Obst, K. R. Brooks, and A. N. Rich, ‘The Müller-Lyer Illusion in a Computational Model of Biological Object Recognition’, *PLoS ONE*, vol. 8, no. 2, p. e56126, Feb. 2013, doi: 10.1371/journal.pone.0056126.

[2] B. Bermond and J. Van Heerden, ‘The Muller-Lyer illusion explained and its theoretical importance reconsidered’, *Biol Philos*, vol. 11, no. 3, pp. 321–338, Jul. 1996, doi: 10.1007/BF00128785.

[3] F. Chollet and others, ‘Keras’, 2015. https://github.com/fchollet/keras