

# Performing the Rorschach test with a neural network

*Version 3.0*

*Coen Dekker*

*Neural Network Project, July 2015*

*Media Technology MSc program,*

*Leiden University*

*Supervisor: Maarten Lamers*

*coen.dekker7@gmail.com*

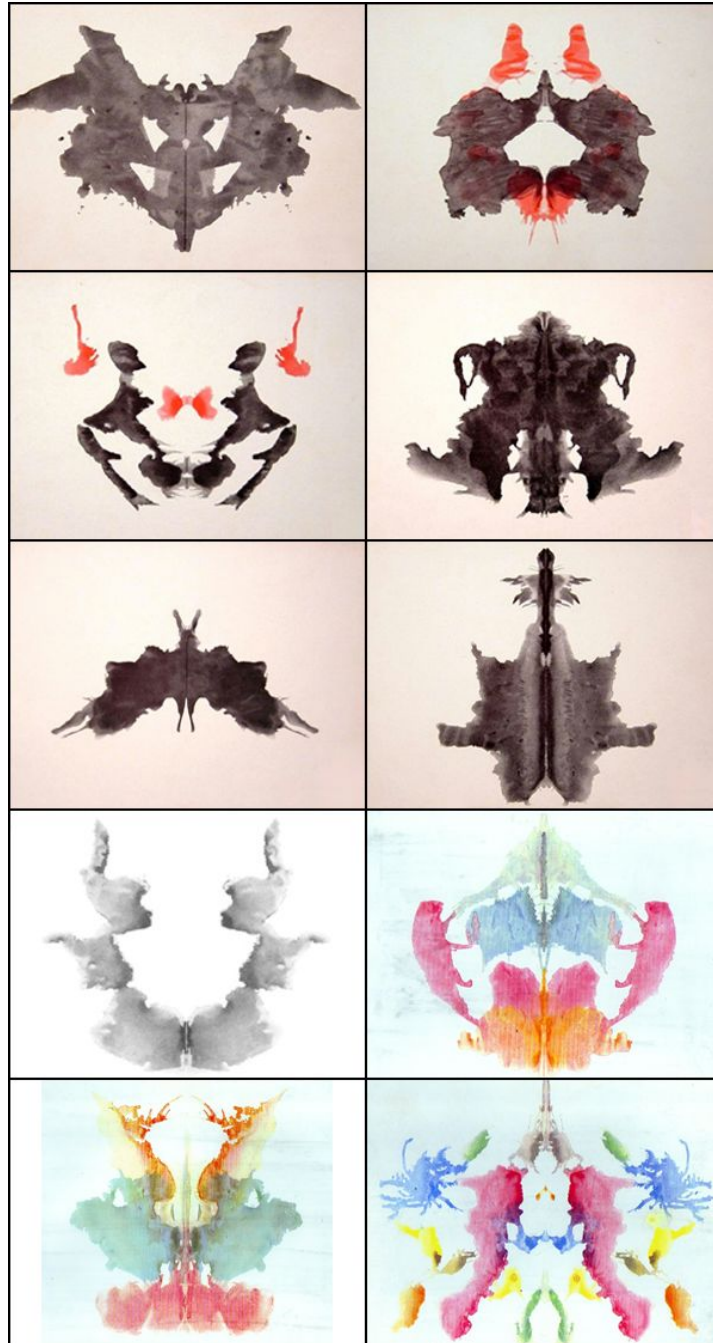
## **Abstract**

In this project a neural network is trained using a specifically designed training set to recognize specific images. The recognition of images is carried out using pixel data of each image. The workload was reduced by PCA. The trained network was then used on Rorschach's Images [4], which were compared to human results of the Rorschach test of Beck [1], Piotrowski [7] and Dana [2]. The results suggest that the three human sources are in line, however the neural network was not comparable to these results since it differs too much.

## **Introduction**

Humans are able to perceive and analyse objects in various shapes and sizes. For example a 4-year-old child can easily categorize animals when looking at pictures of them. Even though it is a two-dimensional image, the child can name different attributes like size, leg count, fur type, eye color etc. The human brain is specialized in categorizing various features so it can ease the job of recognition in daily life.

A psychological test known as the Rorschach test focuses on the participant's perception of abstract cards. The Rorschach test contains a set of ten cards with practically symmetric ink blotted images. The participant perceptions of these ten cards are recorded and then analyzed using psychological interpretation. Some psychologists use this test to examine a person's personality characteristics and emotional functioning. It has been employed to detect underlying thought disorder, especially in cases where patients are reluctant to describe their thinking processes openly [4]. The test is named after its creator, Swiss psychologist Hermann Rorschach. The images of the Rorschach are illustrated in figure 1.



**Figure 1: The ten images used in the Rorschach test**

Recently Beck [1], Piotrowski [7] and Dana [2] have published their results of the Rorschach test. These results include the responses to each card of the Rorschach test. With this information we can get a clear view on what human responses are when interpreting the Rorschach images. This project explores the use of neural networks with a specialized training set to observe if a trained network recognizes the same characteristics as the human responses. The research question for this project is: *Does a trained neural network recognize similar characteristics as humans when looking at Rorschach's images?*

## Related Work

Recently a short study was done by an online resource where the Wolfram Alpha Computational knowledge engine was used to identify the Rorschach images [6]. Each image was used as separate input to interpret its contents. The results of this study show that the Wolfram is using abstract words to describe Rorschach's images, with the exception of the first card seen in figure 1, where the result was: carnivorous bat. It is partly understandable that some images are seen as abstract things since they are not real life examples. The Rorschach test is meant for human interpretation and at least the Wolfram artificial intelligence know most of them are abstract. Words like vertebrate, device, artifact were common answers. Answers like this indicate that there is limited identification of the images. One last interesting result was the answer cinquefoil to the last Rorschach image seen in figure 1. This is a clear answer where the Wolfram artificial intelligence is sure that it is cinquefoil, which is a flower as is presented in figure 2. When looking into the Rorschach image it bears little to no resemblance to cinquefoil. It can be argued that Rorschach's 10th image is hard to interpret since it has several different elements. Still cinquefoil is far off from crab, lobster, spider, rabbit head, caterpillars, worms and snakes, which is what the human answers included.

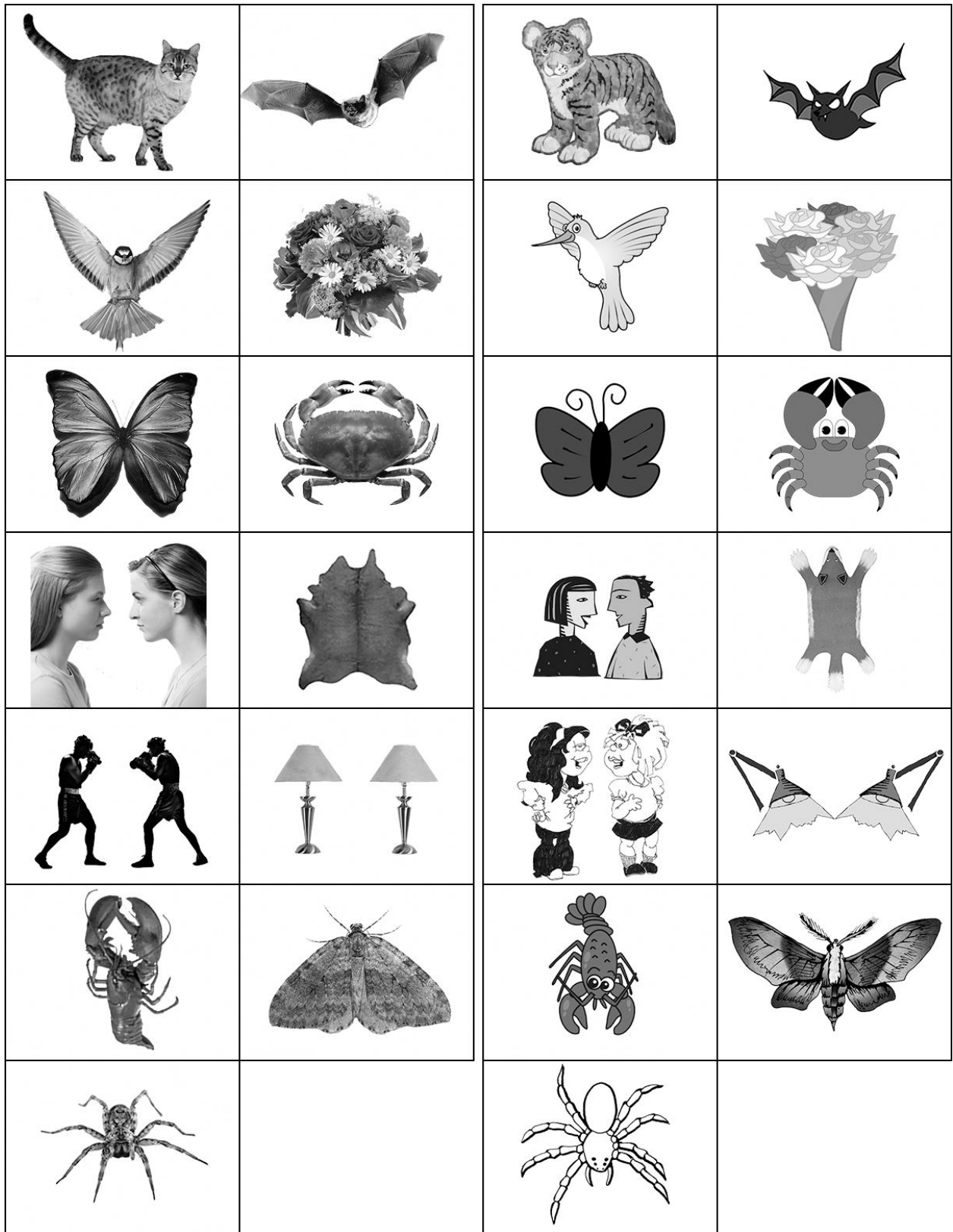


*Figure 2: Comparing cinquefoil (left) to Rorschach card X (right)*

## Method

We started by researching the existing results from the Rorschach test of Beck [1], Piotrowski [7] and Dana [2]. Based on their results we made our own categories which are the basis for our specially developed training set. The categories included: animals (four legged mammals), bats, butterflies, crabs, faces (two human faces), hides (animal hides), humans (two facing each other), lobsters, moths and spiders. As a control group we added a couple additional categories which loosely resemble the existing categories so they could be included in the training set. The control categories included: birds, bouquets and lamps (two lamps facing each other). Now we had a total of thirteen categories. For the thirteen categories we selected ten images on Google Search to get semi-symmetric images of each category. To broaden the learning of the network we decided to add two cartoon-like images to each category, again selected from Google Search. Humans also do not learn solely from real world examples. In the end each category consisted of twelve images, which made a training set of 156 images in total. Since most of the Rorschach images are in a grayscale we also made the images grayscale using Photoshop. A comfortable size of 225 by 300 pixels was chosen to be easy recognizable,

while still being able to compute in a neural network. Figure 3 shows the first and eleventh (cartoon-like) image of all thirteen categories.



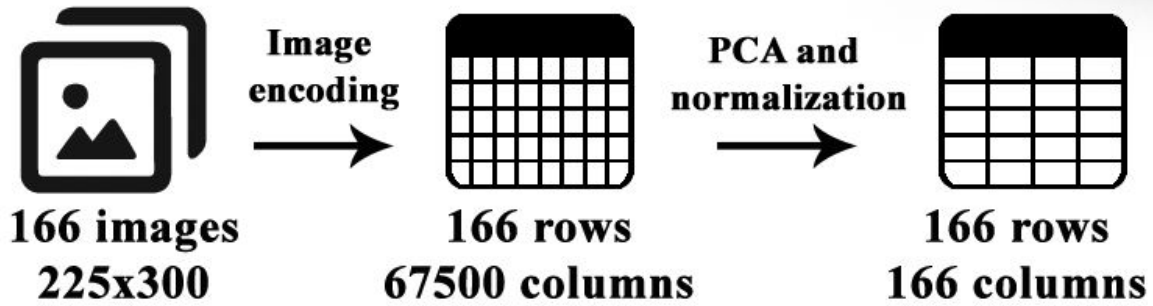
**Figure 3: Image 1 of each category (left), Image 11 of each category (right)**

At this point the training set consisted of 156 images, each 225 by 300 pixels, in JPEG image format. To learn an image to a neural network we decided to encode all the individual pixel colour values, which are 67500 values per image. The converting of images to a data table was done using Processing programming software. The result was a .csv file with the label of each image, followed by 67500 colour values for each image. At this point we could just enter the data table as training set into a neural network in the R language [5]. The problem with that approach is that the workload of a neural network using 67500 input nodes would be too high to compute properly, so instead we chose to run a Principle Component Analysis (PCA) to greatly reduce the needed input nodes.

It should be noted that the current training set has a lot of white background in the images, which is not suitable for a PCA because it requires each column of values to have a little difference because PCA divides by the standard deviation of each column. If one position of an image has the same colour value in all images, the difference will be 0. Dividing by 0 will result in the PCA to malfunction. This was the case a lot of times in our training set because the images had large portions that are only white background, colour value 255. To solve the issue we randomly selected half of the training set to modify the background value from 255 to 254. This little change resulted in a difference, which is so small that it will not matter for our research and it also solves the problem discussed earlier. As we executed the PCA we concluded that keeping all 156 dimensions would be wise since there is no information loss over the training set. To finalize the training set for the PCA we added the Rorschach images, all formatted in the same fashion as the earlier images, in a 225 by 300 grayscale JPEG format. The whole training set now consisted of 166 images. The PCA resulted in a data table of all images labeled followed by 166 numeric values. Upon closer inspection some values far exceeded our expectations, with extreme values like 334 and -578. This was not a direct problem, but we chose to normalize the data to be more compatible with a neural network. This was done by subtracting the mean of each column and dividing by the standard deviation of each column. The normalizing formula is presented below.

$$z = \frac{x - \mu}{\sigma} \quad (1)$$

Where Z is normalized x, x is a datapoint,  $\mu$  is mean of the datacolumn and  $\sigma$  is standard deviation of the datacolumn. The normalizing resulted in a maximum values like 10 and -10, which was more favorable for a training set. The processing overview from images to the normalized data table is presented in figure 5.



*Figure 5: The training set was first made in a grayscale format of 225x300 pixels, next it was analysed using Processing to a table of 166 rows and 67500 columns of pixel values. Finally this data was put into a Principle Component Analysis and normalized to fit a neural network.*

After optimizing the size of the training set it was suitable to work in R. For a neural network we selected an R Package called 'neuralnet' [3]. At this point the ten Rorschach images are excluded from the learning procedure to be used later on. The neural network was trained using different settings summarized in table 1.

**Table 1: Settings used to train neural networks**

	Number of hidden layers	Number of repetitions	learning algorithm	Error function	Linear output
<b>Setting A</b>	30	30	rprop+	sse*	FALSE
<b>Setting B</b>	10	30	rprop+	sse*	FALSE
<b>Setting C</b>	30	100	rprop+	sse*	FALSE
<b>Setting D</b>	30	30	rprop+	sse*	TRUE

## Results

When looking into the results of each setting it is clear that the neural network cannot succeed in recognizing every category. After using the training set with four different settings the same data was tested to evaluate the networks learnability. If the network picks the right category, it was seen as correct. If the network preferred another category or had no definitive answer it was seen as an error. The number of correct answers given and the error rate is presented in table 2.









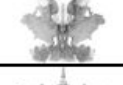

**Table 2: Number of correct answers given and the error rate when testing four trained networks with different settings (settings are illustrated in table 1).**

	Correct answers	Error rate
<b>Setting A</b>	119 of 156	+ - 24%
<b>Setting B</b>	97 of 156	+ - 38%
<b>Setting C</b>	123 of 156	+ - 21%
<b>Setting D</b>	156 of 156	0%

As presented in table 1 and 2, the addition of non linear output (setting D) had a considerable impact on the neural network learning. A 0% error rate suggests overfitting of the network. But using 30 hidden layers and 30 repetitions should not result in overfitting. Therefore setting D was chosen as the main setup to test the Rorschach images.

Since setting D included linear output the values range from minus values to positive values. Negative values only mean "less probable" than 0. An overview of the results for each Rorschach image are presented in table 3.

**Table 3: Results on Rorschach images using setting D (see table 1 and 2 for more information on setting D). The rows are all the Rorschach in order and the columns are the thirteen categories. Results with 0.6 or higher are marked green since they serve as a noteworthy answer.**

	Animal	Bat	Bird	Bouquet	Butterfly	Crab	Faces	Hide	Humans	Lamps	Lobster	Moth	Spider
	-1.31	0.15	0.22	-0.13	0.45	0.41	0.44	0.02	0.71	0.15	1.46	-1.05	-0.53
	-2.22	0.68	1.19	-0.77	-0.51	-0.43	-0.28	-0.27	0.26	1.52	1.79	-0.86	0.88
	1.32	-0.82	0.90	-0.33	0.35	-0.52	1.06	0.48	-0.08	-1.59	1.23	-0.01	-0.97
	-1.42	1.36	2.02	-1.18	-1.11	0.75	-0.29	-0.62	-0.52	0.27	2.24	-0.72	0.21
	1.63	0.25	0.15	-0.17	0.14	-0.77	-0.12	0.77	0.22	-0.65	-0.03	-0.02	-0.40
	-0.91	0.45	0.89	-0.66	-0.83	0.31	0.09	0.15	-0.81	0.26	1.34	-0.56	1.28
	-2.54	0.95	2.38	-1.42	-1.70	0.10	-1.05	-0.54	1.26	1.66	2.90	-2.26	1.26
	-0.80	0.22	1.40	-0.82	-0.81	-0.01	-0.44	-0.49	1.30	1.03	0.93	-1.19	0.67
	-1.79	0.75	2.52	-0.62	-2.13	-1.42	-1.39	-0.81	1.73	0.94	3.48	-2.80	2.54
	-2.50	0.32	0.34	0.10	-0.28	-0.17	-0.78	-0.75	0.65	2.29	1.13	-1.26	1.91

The results are not promising. Just like the Wolfram study these results are all over the place. When inspecting the results per image there is not a single image that relates to the results of Beck, Piotrowski and Dana. There is no correlation found when looking into these results, suggesting that the network did not recognize the images like humans do.

Table 4 shows the results of Beck, Piotrowski and Dana combined. These results are closely comparable since they share much of the same answers. The results are not in line with the trained neural network.



**Table 4: comparing the results of Beck (first column per category), Piotrowski (second column per category) and Dana (third column per category). Gradings made using the resource material, if there is no grading of multiple options, multiple 1's were added to the table.**

Gradings Beck, Piotrowski & Dana													
	animals	bats	birds	bouquets	butterflies	crabs	faces	hides	humans	lamps	lobsters	moths	spiders
rorschach 1		1 1			1 2 1							1	
rorschach 2	1 1								1				
rorschach 3									1 1 1				
rorschach 4								1 1 1					
rorschach 5		1 2 2			1 1 1							1	
rorschach 6								1 1 1					
rorschach 7							1 1 1						
rorschach 8	1 1 1												
rorschach 9									1				
rorschach 10						1					1		1

## Conclusion

The neural network did not succeed in mimicking human answers of the Rorschach test. All the measurements taken to reduce the workload of the neural network did not reduce the information that it contained, rather it was in a more suitable format to compute in a neural network. Using the R package 'neuralnet' resulted in only one setting to be acceptable, without overfitting the neural network. Still it did not show any significant answers that we can relate to. An argument which might explain this result is that the Rorschach images are highly abstract, where the images are still realistic. Even with the addition of the cartoon images it might not be a suitable training set to recognize Rorschach images.

Another argument which might explain this result is that all the images are converted to grayscale, even though the Rorschach images eight, nine and ten are originally in colour. This only affects the last three images, but changing the colours could greatly influence the interpretation of the images, which makes them different from the original Rorschach test.

A third argument which might explain this result is the use of the neuralnet package. For the limited time of this research it was the only one that was thoroughly tested. Only one setting was actually used in the final experiment. Changing packages or even software might change the way a neural network is trained, which might indicate why this result was achieved and the reason for it not being suitable.

One final argument we would like to point out is that the chosen categories might differ more from the Rorschach images as originally intended. We chose to select multiple bats in different perspectives to give the network a general way of recognizing bats. In Rorschach image five, which closely resembles a bat, only a seemingly top down view was used. Our aim to train the network on a general level of recognizing bats, or any other category, might not be the right way to recognize Rorschach's images.

Of course it might also be the case that neural networks will never see the same as the human results do.

## **Future Work**

This research is executed on a highly specific setting. It is unfortunate that the trained network showed no sign of identification of the Rorschach images. With the existing training set it would be convenient to continue working on this research using different neural network packages and settings to hopefully observe if that would make a difference.

## **Acknowledgements**

I would like to thank Maarten Lamers for the idea of this research and his advice and comments during the research.

## **References**

1. Burstein, A. G., & Loucks, S. (1989). Rorschach's test: scoring and interpretation. Taylor & Francis.
2. Dana, R. H. (Ed.). (2000). Handbook of cross-cultural and multicultural personality assessment. Routledge.
3. Fritsch, S., Guenther, F., & Guenther, M. F. (2012). Package 'neuralnet'. Training of Neural Network, 1.
4. Gacono, C. B., & Meloy, J. R., (2013), The Rorschach assessment of aggressive and psychopathic personalities. Routledge.
5. Murdoch D., Plummer M., (2015), The R Project for Statistical Computing, <http://www.r-project.org/>
6. Online Resource, (2015), Wolfram Language Image Identification Project takes the Rorschach test, <https://imgur.com/a/tJ9LR>
7. Piotrowski, Z. A. (2013). Perceptanalysis: The Rorschach Method Fundamentally Reworked, Expanded and Systematized. Routledge.