

# Explainable AI für die Bildverarbeitung

Robert Haase

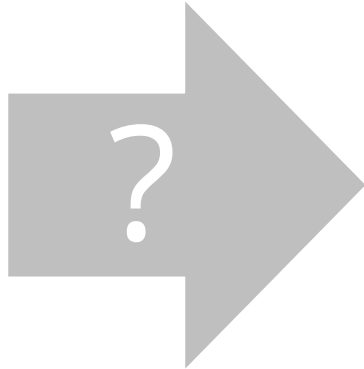


These slides can be reused under the terms of the [CC-BY4.0](https://creativecommons.org/licenses/by/4.0/) license.

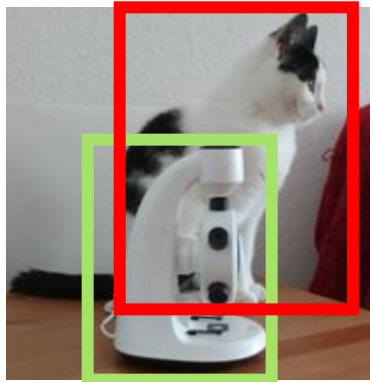
<https://doi.org/10.5281/zenodo.14996127>

# Bild-Klassifikation

**Quiz:** Was könnte ein Ergebnis von Bild-Klassifikation sein?



Katze: 30%  
Mikroskop: 20%  
Hund: 5%  
Auto: 0%  
...



“Das Bild zeigt eine Katze, die neben einem Mikroskop sitzt”

# Pixel-Klassifikation

**Quiz:** Wie bezeichnet man diesen Task?



Kombinatorische  
Segmentierung



Semantische  
Segmentierung



Instanz-  
segmentierung



Connected-Component  
Segmentierung



# Explainable Artificial Intelligence (XAI)

- “Es gibt derzeit noch keine allgemein akzeptierte Definition von XAI.”  
Wikipedia [1]
- Aktives Forschungsfeld

## Relevante Aspekte:

- Menschen ermöglichen
  - KI-Systemen zu vertrauen,
  - mit KI-Systemen effektiv umzugehen und
  - Ergebnisse von KI-Systemen vorherzusagen.

# Explainable Artificial Intelligence (XAI)

Was ist erklärbar?

KI Algorithmus

Entscheidungsfindung  
in einem KI-System

Beitrag zu Grunde  
liegender Daten

*Bedeutung* zu Grunde  
liegender Daten

Wie ist es erklärbar?

Lesen+Verstehen von  
Gleichungen und Quellcode

Visualisierung von  
Zwischenergebnissen

Messung von Relation zwischen  
Eingangs- und Ergebnisdaten

# Erklärbarkeit

Eine in sich schlüssige Argumentationskette die einen Sachverhalt, oder einen Algorithmus vollständig transparent abbildet.

## Intrinsisch erklärbare KI-Algorithmen

- Beispiel: Lineare Regression

$$f(x_1, x_2) = w_1 x_1 + w_2 x_2$$

Wenn  $w_1$  viel größer ist als  $w_2$ , ist das Gesamtergebnis stark von  $x_1$  abhängig und der Einfluss von  $x_2$  eher gering.

Modell  
erklärbar

Ergebnisse  
vorhersagbar

# Erklärbarkeit

Eine in sich schlüssige Argumentationskette die einen Sachverhalt, oder einen Algorithmus vollständig transparent abbildet.

## Intrinsisch erklärbare KI-Algorithmen

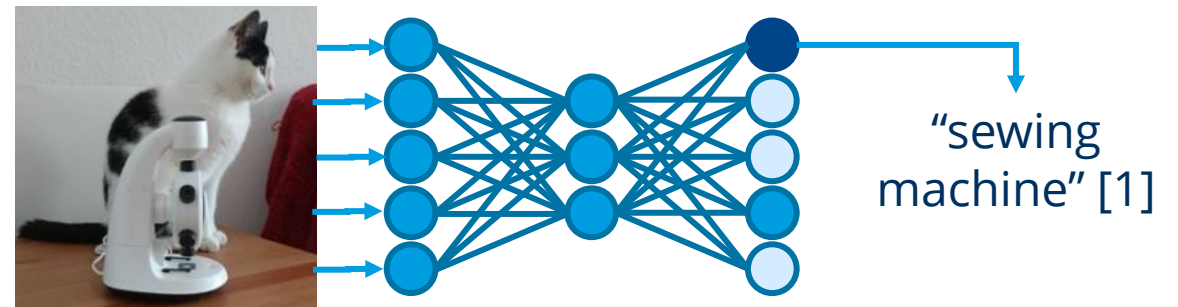
- Beispiel: Lineare Regression

$$f(x_1, x_2) = w_1 x_1 + w_2 x_2$$

Wenn  $w_1$  viel größer ist als  $w_2$ , ist das Gesamtergebnis stark von  $x_1$  abhängig und der Einfluss von  $x_2$  eher gering.

## Black-Box KI-Algorithmen

- Beispiel: Deep Neural Networks (DNN)



Nicht ohne Weiteres erklärbar und  
Ergebnisse auch nicht solide vorhersagbar

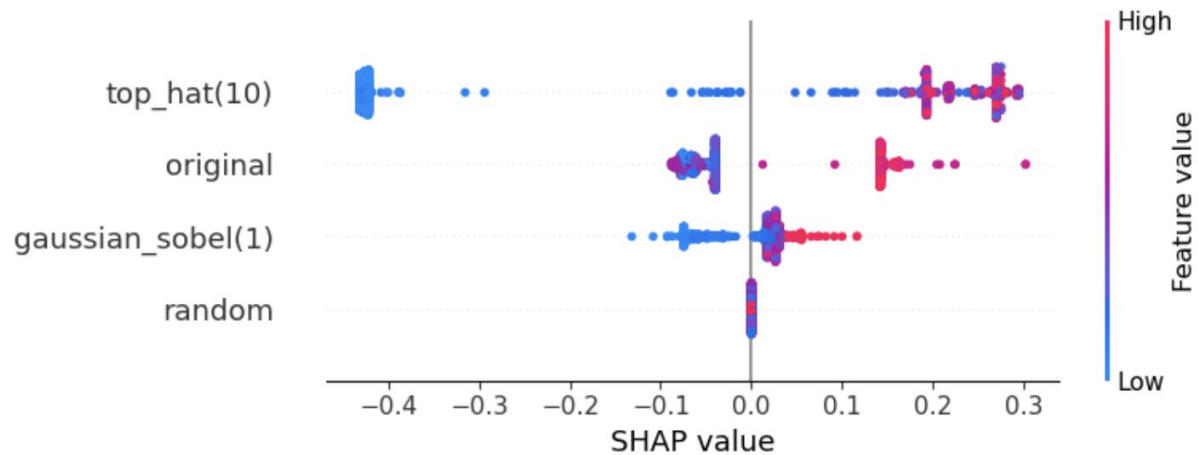
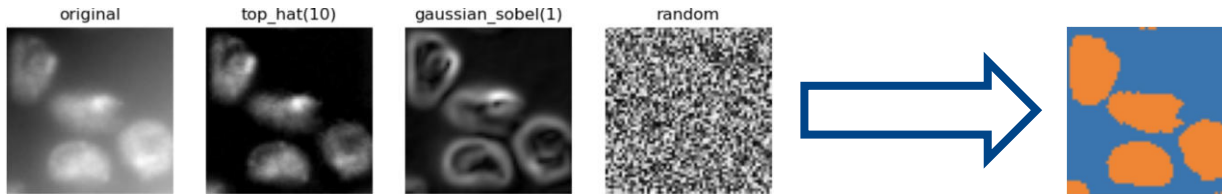


# Interpretierbarkeit

## Visualisierung von Zwischenergebnissen und deren Einfluss auf Ergebnisse

### Modell-agnostische Methoden

Beispiel: Shapley's Additive exPlanations (SHAP)



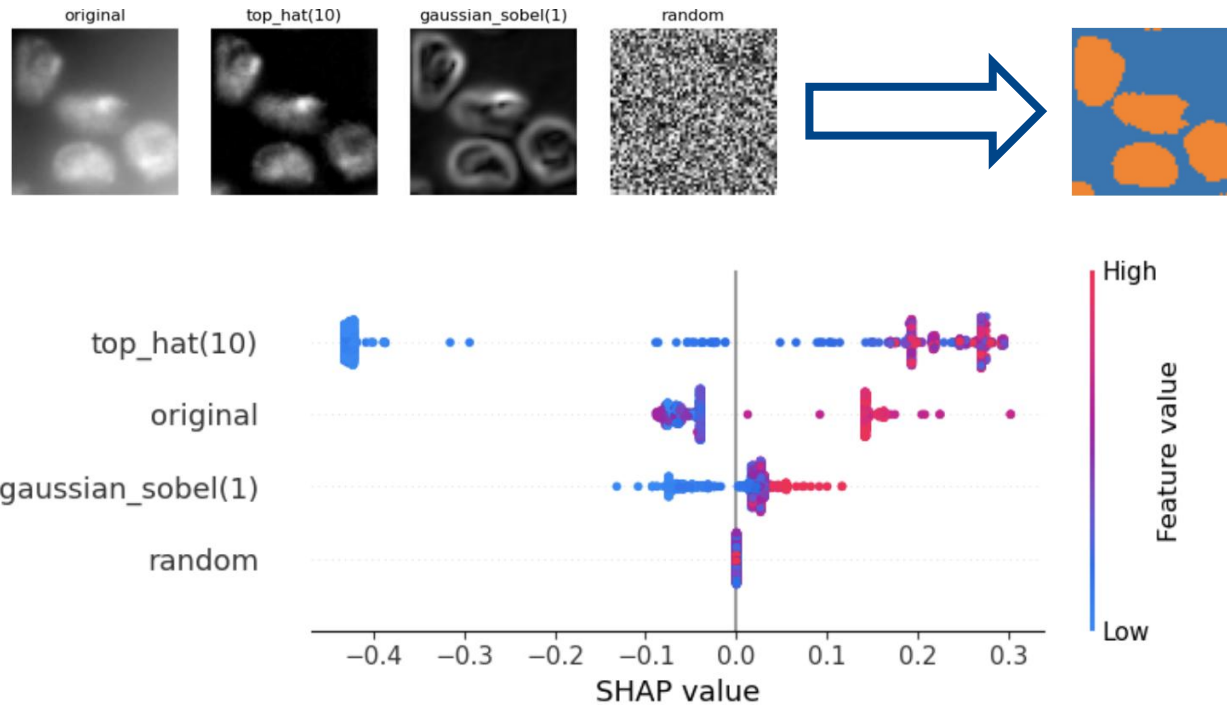


# Interpretierbarkeit

## Visualisierung von Zwischenergebnissen und deren Einfluss auf Ergebnisse

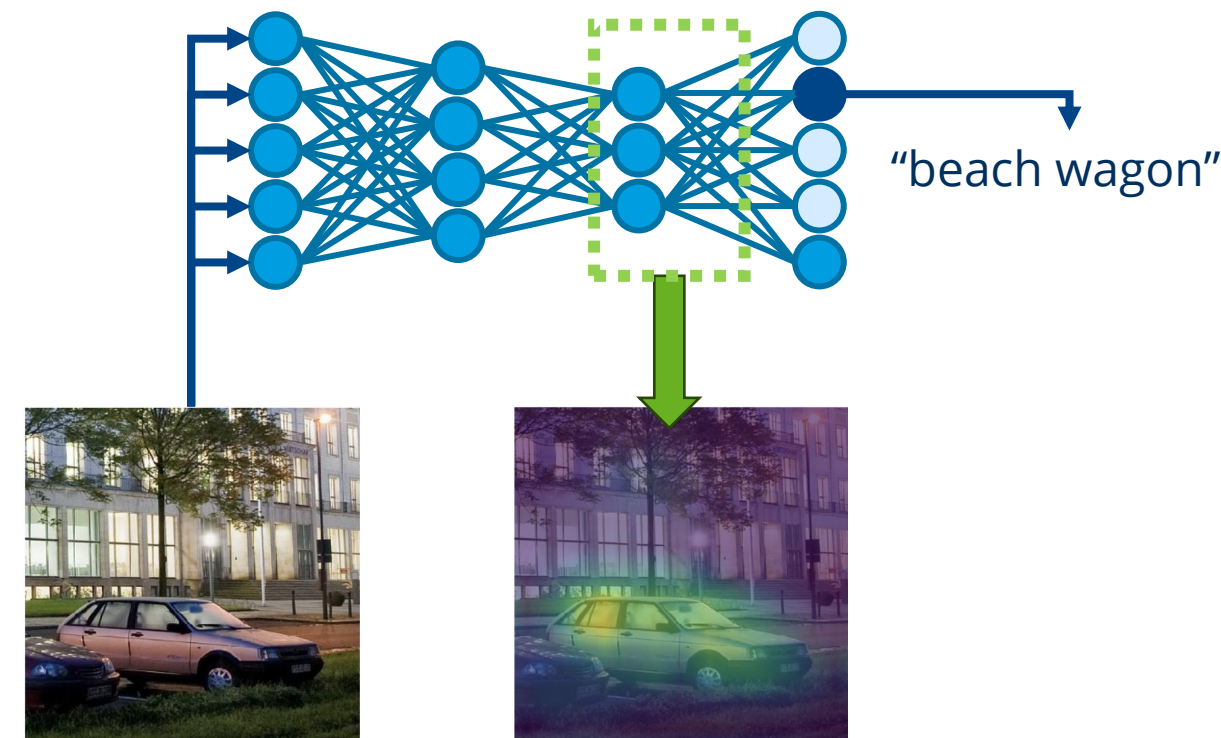
### Modell-agnostische Methoden

Beispiel: Shapley's Additive exPlanations (SHAP)



### Modell-spezifische Methoden

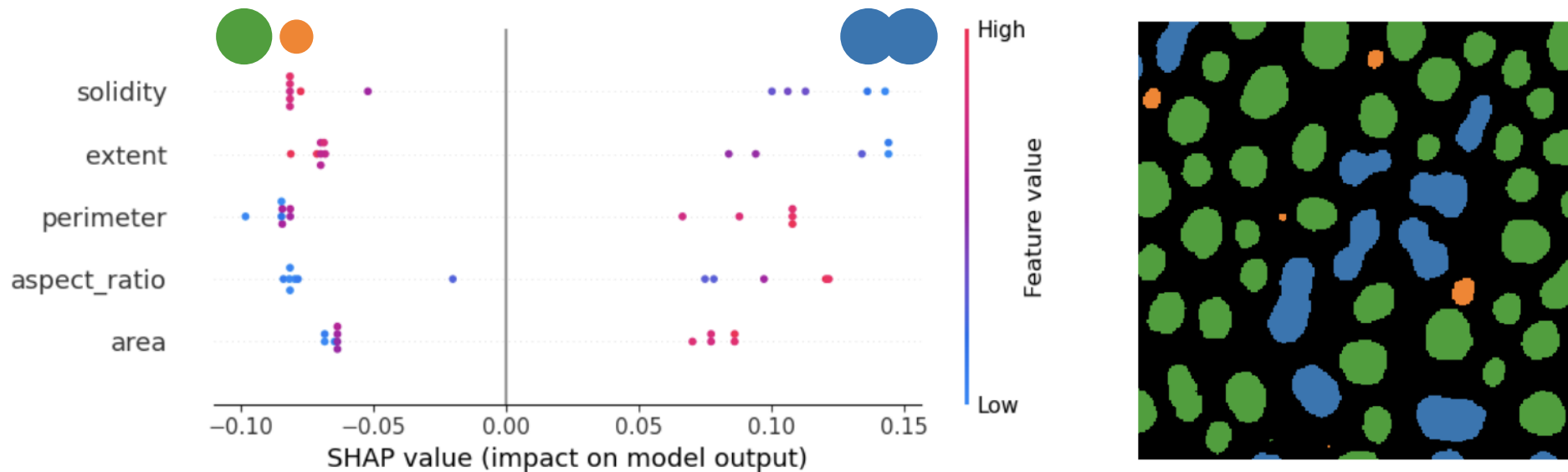
Beispiel: Gradient Class Activation Maps (Grad-CAM)



# Zielgruppen

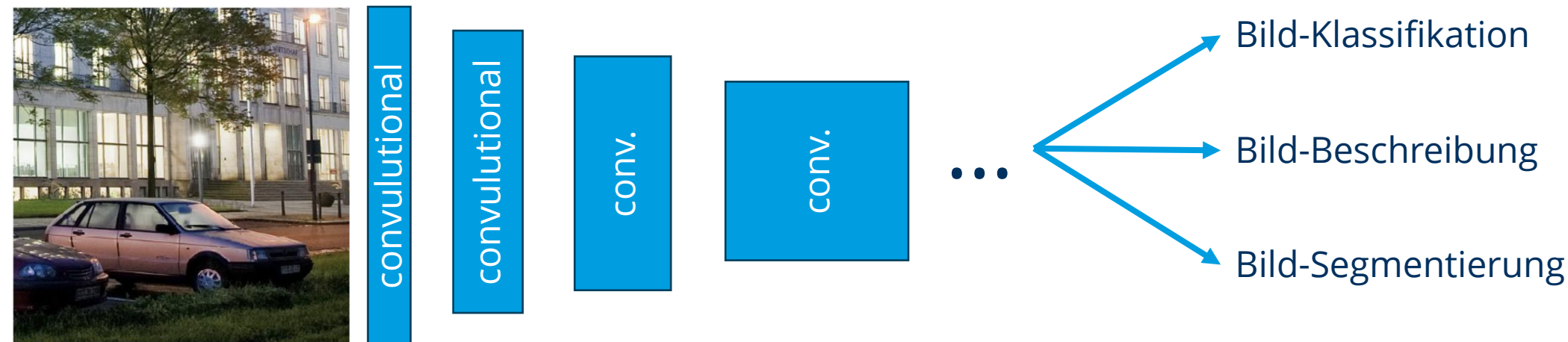
Ja nach Zielgruppe [für die Erklärung] sind Einfluss von Daten wichtiger als Funktionsweise von KI-Algorithmen.

- Viele Informatiker:innen wollen KI-Methoden erklären und verstehen.
- Biolog:innen nutzen KI als Methode um biologische Prozesse zu erklären.
- Beispiel: "Welche Parameter unterscheiden **runde Objekte** von **länglichen**?"



# Gradient Class-Activation Maps (Grad-CAM)

- Funktioniert nur mit NN-Algorithmen, die Eingangsdaten zunächst mit Convolutional Layers bearbeiten. (Modell-spezifisch)
- Unabhängig von rechter Hälfte des NNs (Modell-agnostisch)
- Visualisiert Zwischenergebnisse umd Entscheidungsfindung im KI-System interpretierbar zu machen



# Gradient Class-Activation Maps (Grad-CAM)

Wird auf bestehendes Netzwerk angewandt; keine Anpassung der Architektur notwendig (post-hoc Methode).

Input image

Convolutional layers of a DNN such as ResNet

Output: a vector of probabilities.



convolutional

convolutional

conv.

conv.

- 0.7 Beach wagon
- 0.1 goldfish
- 0.1 palace



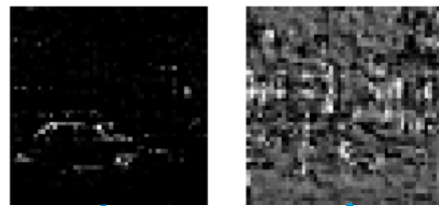
# Gradient Class-Activation Maps (Grad-CAM)

Wird auf bestehendes Netzwerk angewandt; keine Anpassung der Architektur notwendig (post-hoc Methode).

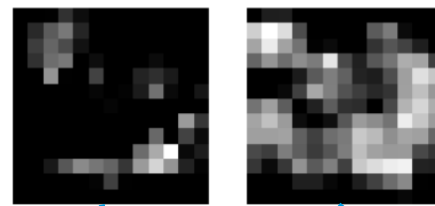
Layer 1 (256, 100, 100)



Layer 2 (512, 50, 50)



Layer 4 (2048, 13, 13)



“2028 Feature-Bilder  
mit je 13x13 Pixeln”

400x400



convolutional

convolutional

conv.

conv.

- Beach wagon
- goldfish
- palace

# Quiz

Wie nennt man typischerweise **diesen Teil** eines DNNs?

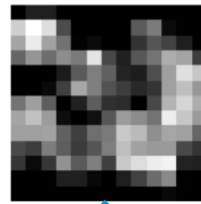
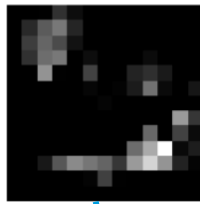
Layer 1 (256, 100, 100)



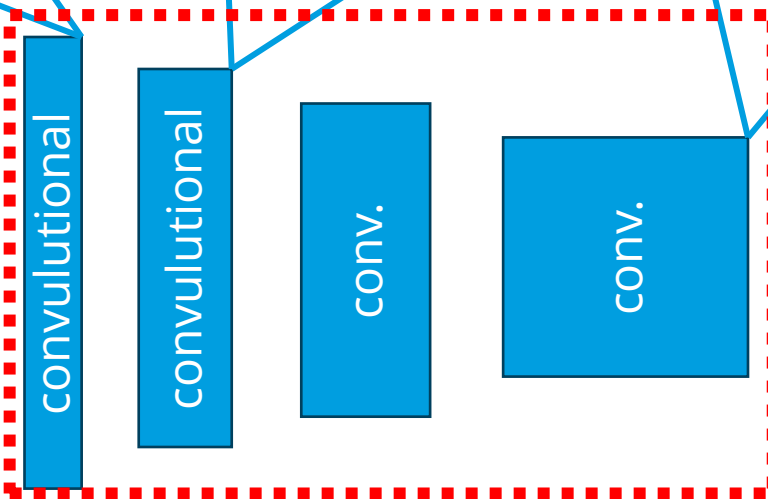
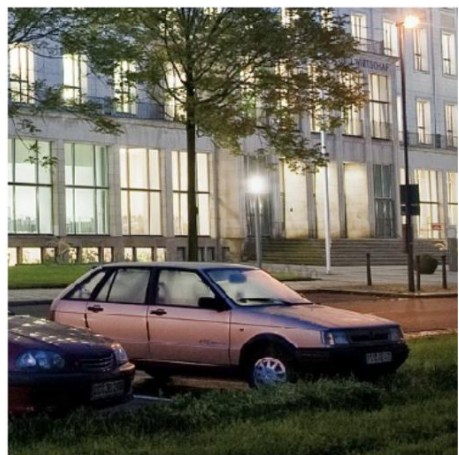
Layer 2 (512, 50, 50)



Layer 4 (2048, 13, 13)



400x400



- Beach wagon
- goldfish
- palace

Reducer



Increaser



Encoder

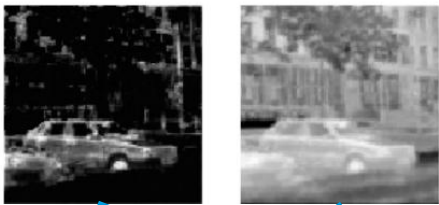


Decoder

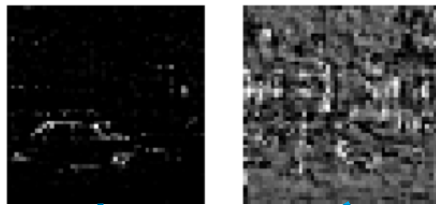


# Gradient Class-Activation Maps (Grad-CAM)

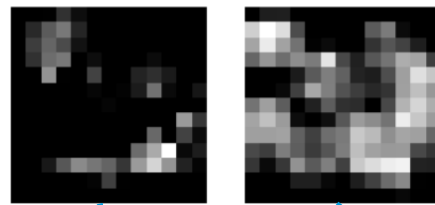
Layer 1 (256, 100, 100)



Layer 2 (512, 50, 50)



Layer 4 (2048, 13, 13)



Keines dieser Bilder sagt direkt etwas über Bildinhalt aus. Es gibt kein Feature Image "Beach wagon"

400x400



convolutional

convolutional

conv.

conv.

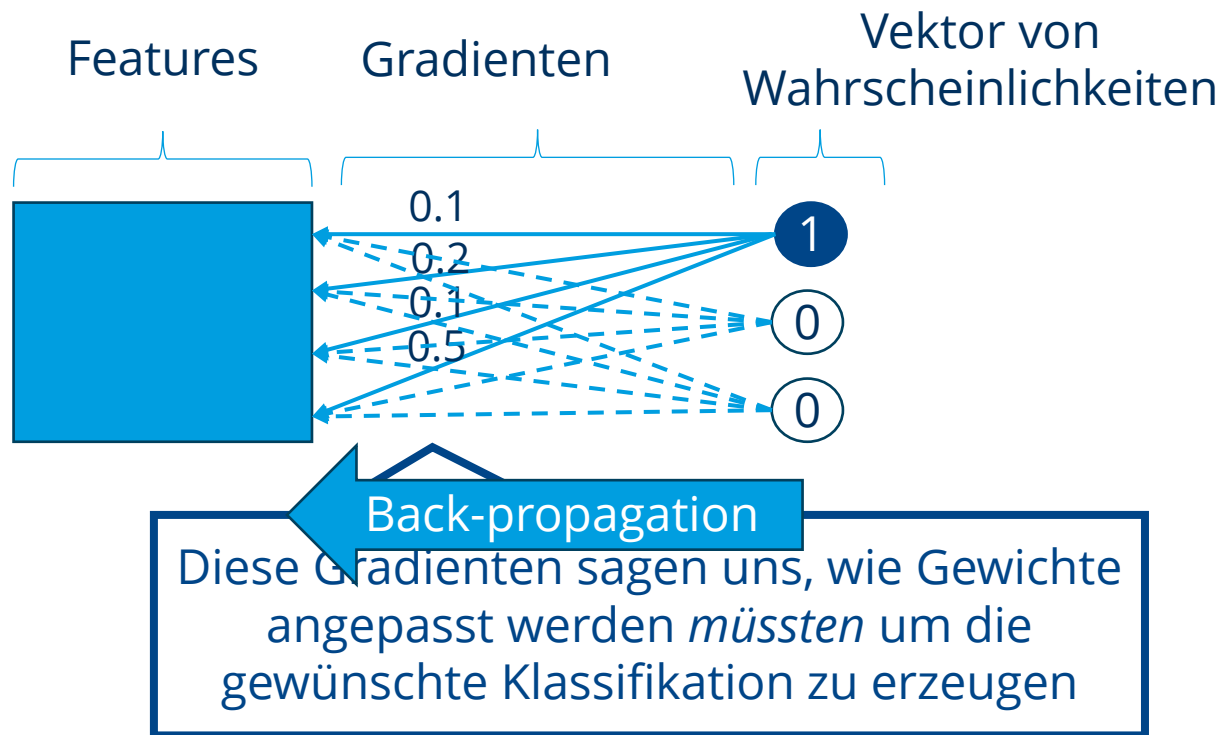
- Beach wagon
- goldfish
- palace

Grad-CAM passiert hier



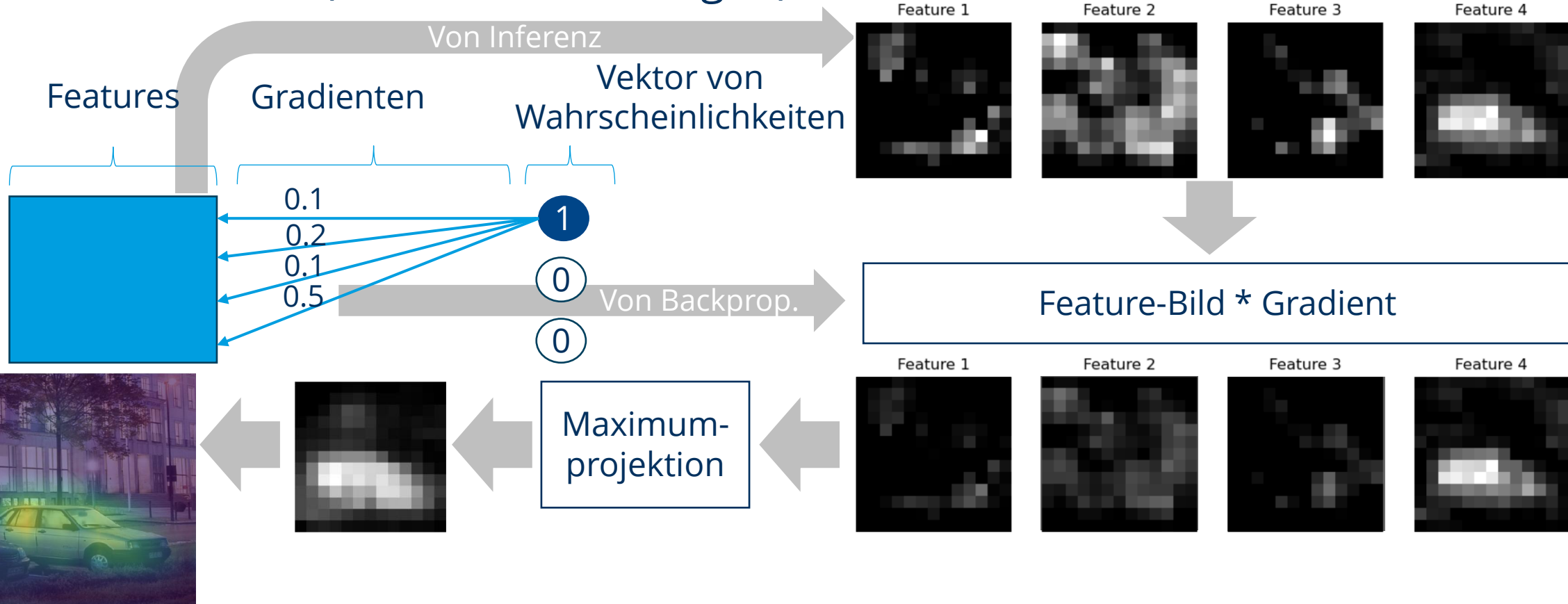
# Gradient Class-Activation Maps (Grad-CAM)

Back-Propagation einer perfekten Klassifikation (1,0,0) gibt uns Gradienten (Gewichtsänderungen) um die Klassifikation zu verbessern.



# Gradient Class-Activation Maps (Grad-CAM)

Back-Propagation einer perfekten Klassifikation (1,0,0) gibt uns Gradienten (Gewichtsänderungen) um die Klassifikation zu verbessern.



# Gradient Class-Activation Maps (Grad-CAM)

Back-Propagation einer perfekten Klassifikation (1,0,0) gibt uns Gradienten (Gewichtsänderungen) um die Klassifikation zu verbessern. Das funktioniert auch mit anderen möglichen Klassifikationen bspw (0,1,0).

“beach waggon”



“palace”



“flagpole”



“great white shark”



# Quiz

Angenommen, dieser Layer hat  $2048 \times 13 \times 13$  Ausgaben. Wofür steht die 2048?

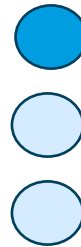


convolutional

convolutional

conv.

conv.



Anzahl  
Features



Anzahl der  
Klassen



Breite der  
Feature-Bilder



Anzahl  
Layer





# Quiz

Angenommen dieser Vektor hat 1000 Elemente. Wofür steht die 1000 ?



convolutional

convolutional

conv.

conv.



Anzahl Features



Breite der Feature-Bilder



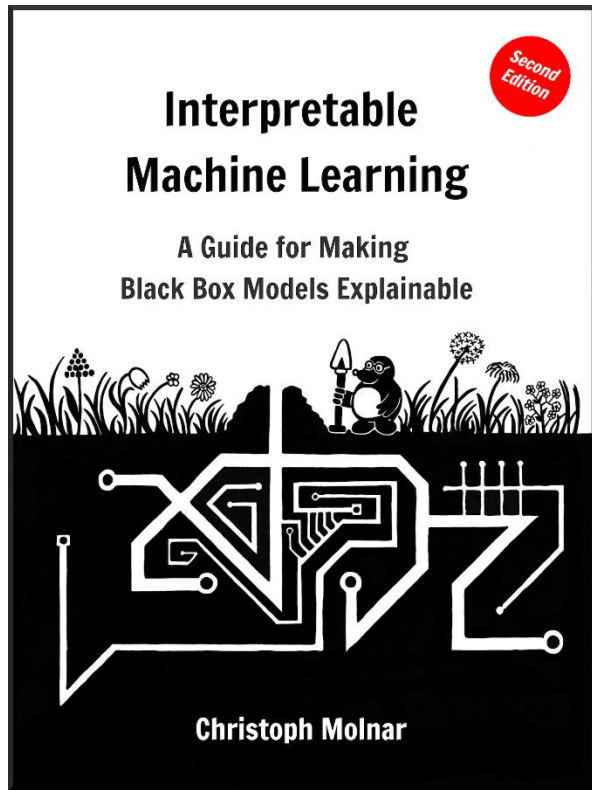
Anzahl der Klassen



Anzahl Layer



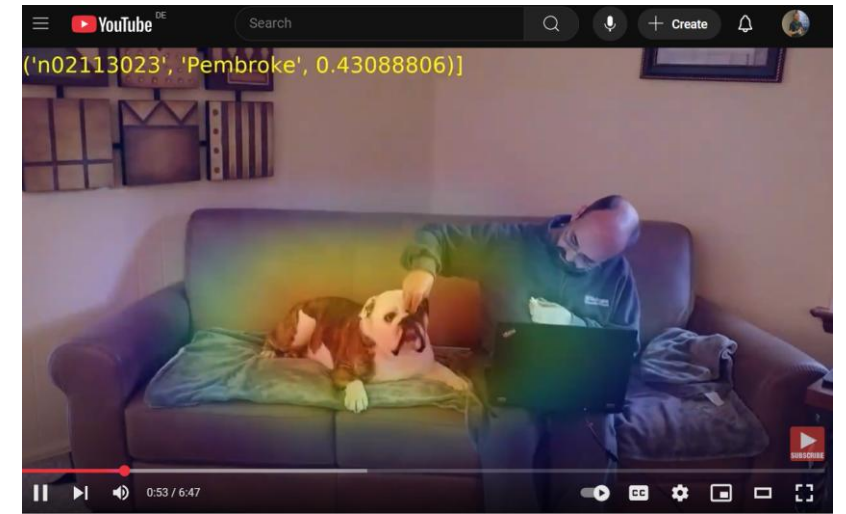
# Read more...



<https://christophm.github.io/interpretable-ml-book/>



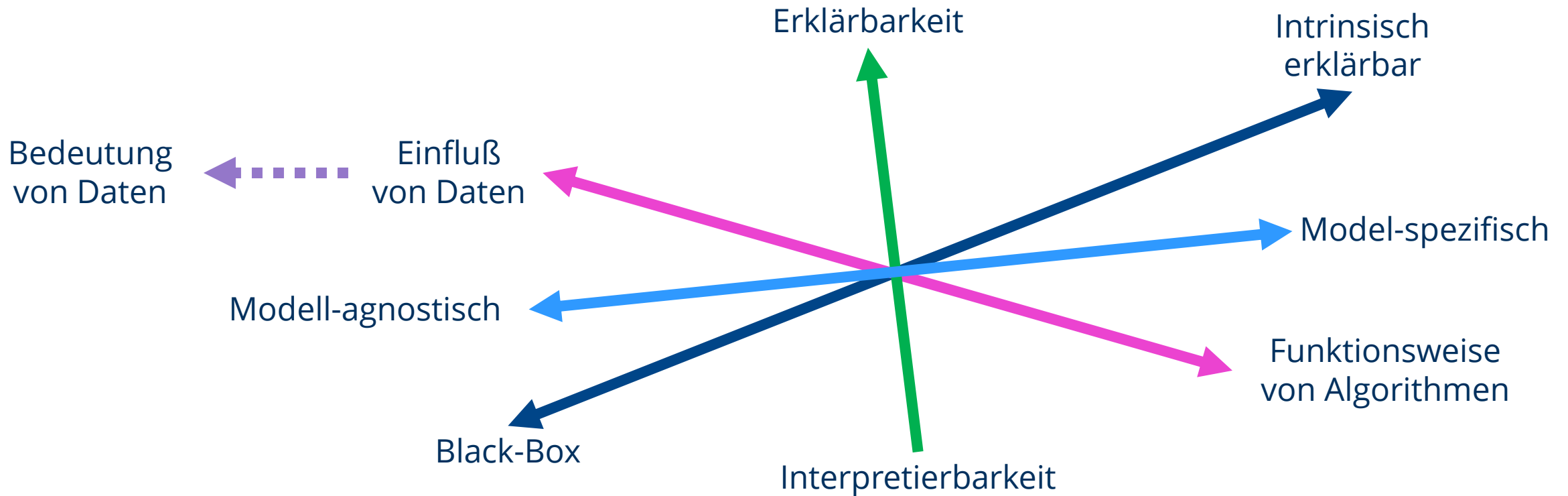
<https://www.amazon.de/dp/3030686396>



[https://www.youtube.com/watch?v=dw63QH\\_b3Jo](https://www.youtube.com/watch?v=dw63QH_b3Jo)

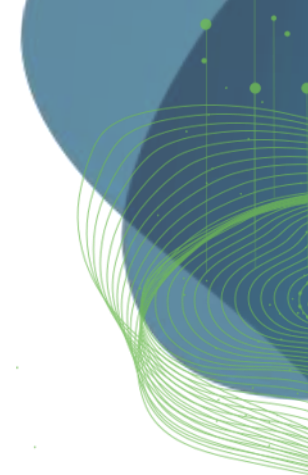
# Zusammenfassung: Explainable AI

Methoden der XAI kann man auf unterschiedlichen Skalen einordnen





# Exercises



# Exercises

Explaining Object classification

haesleinhuepf.github.io/xai/30\_shap/object\_classification.html

ScaDS.AI  
DRESDEN LEIPZIG

Search th [Ctrl] [K]

Explainable Artificial Intelligence Notebooks

Setup  
Setting up your computer

SHAP Analysis  
Pixel classification explained with SHAP  
Explaining Object classification using SHAP

Grad-CAM  
Gradient Class-Activation Maps (Grad-CAM)

Links  
Imprint

### Explain classification using SHAP values

```
# Import necessary Libraries
import shap

# Calculate SHAP values
explainer = shap.TreeExplainer(rf)
shap_values = explainer.shap_values(X)[...,0]

shap.summary_plot(shap_values, X) #, feature_names=feature_columns)
```

Exercise

Draw the SHAP summary plot for the shap values [..., 1]. Which object class was this SHAP plot drawn for?

## SHAP-values

Gradient Class-Activation Maps

haesleinhuepf.github.io/xai/60\_grad-cam/classification\_resnet.html

ScaDS.AI  
DRESDEN LEIPZIG

Search th [Ctrl] [K]

Explainable Artificial Intelligence Notebooks

Setup  
Setting up your computer

SHAP Analysis  
Pixel classification explained with SHAP  
Explaining Object classification using SHAP

Grad-CAM  
Gradient Class-Activation Maps (Grad-CAM)

Links  
Imprint

```
show_cam_for_class("great white shark")
```

### Exercise #

What needs to be changed above to make sure the classification returns "car"?

### Exercise

Write a Python function that takes an image filename as parameter and returns the class name as string and a corresponding CAM image. Call this function in a loop which iterates over all images in the folder 'data'.

## Grad-CAM