Classifying Real Estate Ad Images with Convolutional Neural Networks
CAS Machine Intelligence, Deep Learning, April 2018

School of Engineering — ZHAW Zürcher Hochschule für Angewandte Wissenschaften

Introduction & Motivation

- On real estate portals one often finds pretty bad images; even the first picture is often not nice.
- We used a dataset of 2'000 real estate ad images to build a classifier, which might help to automatically select an appropriate image (e.g. exterior view for first picture).
- By extracting the EXIF-Tags the images were grouped into 6 classes (exterior view, interior view, kitchen, bathroom, floor plan, other).
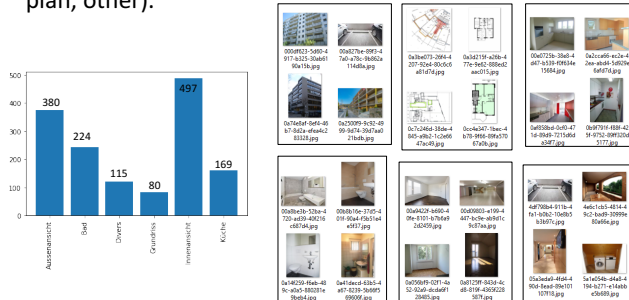
Results

- Overall the best model achieved an accuracy on the test set of 0.93 (~200 images). However this model used only three of the 6 classes.
- On all 6 classes the pretrained VGG16 model achieved the best accuracy with 0.84.
- Classification reports provides further details to the performance of the different classes:

Model 4: 5 Conv. Layers (3 classes) **Model 5: Retrained VGG16 (all 6 classes)**

Classification report - 3 Classes

	precision	recall	f1-score	support
Class 0	0.97	0.81	0.88	73
Class 1	1.00	0.91	0.95	11
Class 2	0.92	0.99	0.95	168
avg / total	0.94	0.93	0.93	252

Models

- We played around and tried different approaches (own models, pretrained features model VGG16) and worked with the keras data generator.
- For all models we used Batchnorm- and Dropout-Layers, and for optimization Adam with cross entropy loss.

| Model	Pretrained	# Classes	# Params	# Conv.-Layers	Accuracy
Model 1	No	6	6,378,314	5	0.75
Model 2	No	6	10,729,526	5	0.70
Model 3	No	5 (w/o other)	10,779,713	5	0.77
Model 4	No	3 (interior, exterior, floor plan)	6,333,219	5	0.93
Model 5	Yes (VGG16)	6	15,936,134 (1'221'446 trainable)	16	0.84

- We decided to further look at Model 4 because we liked the good results without pretraining.

Conclusions

- Models with fewer classes showed higher accuracy and seem ready for real world application.
- More training data is needed for a robust model with more classes.
- Data cleansing should be applied to further improve performance (misclassifications reveal mislabeled training data).

- Imagesize can be fairly small without any loss in accuracy, results were comparable for input imagesizes of 224x224px and 100x100px.
- Using a GPU (on google colab) speeds up training by a factor of 20 (180s/epoch vs. 9s/epoch to train Model 5 (VGG16)).

Insights

- The visualized filters/feature maps show the basic structures that were learned by the model; the last layer seems to make much sense and can be visually linked to the classes.
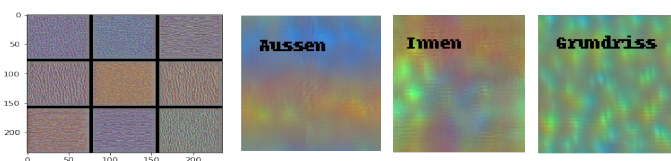
- By visualizing important pixels for the classes interior view, exterior view and floor plan, we can see that it's not just one area that's relevant, but pixels all over have an impact.
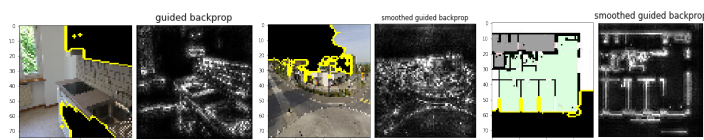
Outlook

- The classification by room/view could be combined with other image classifiers.
- For example the most appealing image could automatically be selected to be displayed first in an advertisement.
- Another possibility would be to detect objects in the images (e.g. dishwasher, fireplace). This information could be used to augment an ad with additional information.
- A combination of all models could be applied in a fully automated online real estate ad process.

References: https://github.com/tensorchiefs, https://github.com/keras-team/keras, https://raghakot.github.io/keras-vis, https://github.com/experiencor/deep-viz-keras, https://github.com/marcotcr/lime, https://www.trulia.com/blog/

Authors: Reto Camenzind, Lukas Stöcklin, Julia Sulc