PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA

DEEP NEURAL NETWORK MODELS WITH EXPLAINABLE COMPONENTS FOR URBAN SPACE PERCEPTION.

ANDRÉS CÁDIZ VIDAL

Thesis submitted to the Office of Research and Graduate Studies
in partial fulfillment of the requirements for the degree of
Master of Science in Engineering

Advisor:
HANS LÖBEL

Santiago de Chile, July 2020

PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA

DEEP NEURAL NETWORK MODELS WITH EXPLAINABLE COMPONENTS FOR URBAN SPACE PERCEPTION.

ANDRÉS CÁDIZ VIDAL

Members of the Committee:
HANS LÖBEL
PATRICIO DE LA CUADRA
MEMBER B
MEMBER C

Thesis submitted to the Office of Research and Graduate Studies
in partial fulfillment of the requirements for the degree of
Master of Science in Engineering

Santiago de Chile, July 2020

Gratefully to my parents and siblings

ACKNOWLEDGEMENTS

Write in a sober style your acknowledgements to those persons that contributed to the development and preparation of your thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The abstract must contain between 100 and 300 words. The abstract must be written in English and Spanish. In the case of doctoral theses, the layout of the abstract page is different, so please check the template provided by the OGRS.

Keywords: thesis template, document writing, **(Write here the keywords relevant and strictly related to the topic of the thesis)**.

RESUMEN

El resumen debe contener entre 100 y 300 palabras. El resumen debe ser escrito en inglés y español. En el caso de tesis de doctorado, el formato de la página del resumen es distinta, por favor verifique la plantilla entregada por la Dirección de Postrgrado.

Palabras Claves: plantilla de tesis, escritura de documentos, **(Colocar aquí las palabras claves relevantes y estríctamente relacionadas al tema de la tesis)**.

1. INTRODUCTION

Urban perception is a feeling held by people about a location. These feelings can be and are often related to a particular characteristic, like happiness or beauty, or also inherently negative ones, like insecurity or fear (Ordonez & Berg, 2014). Understanding the cause of these feelings is a complex task, since unique social and psychological aspects of each individual affect how they perceive and the spaces they observe (Nasar, 1990).

Visual urban perception is responsible for a large parte of the experience that people go through while being at or using an urban space. This not only affects how much the spaces themselves are used (Khisty, 1994) but also the use of related means of transport (Antonakos, 1995). Other studies have also found correlations between urban perception, crime statistics (Ordonez & Berg, 2014) and wealth, and therefore used it as a proxy measure of inequality (Ordonez & Berg, 2014; Salesses, Schechtner, & Hidalgo, 2013; Rossetti, Lobel, Rocco, & Hurtubia, 2019).

Additionally, being able to measure a community's need and perception of a city at scale is something of key importance on developing cities. The insights can be applied for design of public policy so that local governments can allocate their resources more efficiently (Santani, Ruiz-Correa, & Gatica-Perez, 2018).

Traditional methods for obtaining this information, consist of hand made polls about specific locations, making systematic quantification of perception an extremely costly and hard to escalate task (Nasar, 1990; Clifton & Ewing, 2008). An alternate approach consists of surveys based on computer generated images of simulated spaces. This scheme is more scalable, but is limited to experimental design and cannot be directly applied to real urban spaces. (Laing et al., 2009; Iglesias, Greene, & Ortúzar, 2013).

Currently, thanks to the great volumes of data generated by web platforms (Salesses et al., 2013) and to modern deep learning (DL) and computer vision techniques (LeCun, Bengio, & Hinton, 2015), new solutions for quantifying urban perception at scale have

become feasible. The Place Pulse 2.0 dataset (Dubey, Naik, Parikh, Raskar, & Hidalgo, 2016), is the most significant example of this, consisting of pairs of images along with labels that indicate which of the images is more representative of a particular attribute. Previous studies have achieved significant results with it, either by applying traditional deep learning (Dubey et al., 2016) or by combining it with other approaches (Rossetti et al., 2019; F. Zhang et al., 2018). In general, works trying to quantify urban perception at scale consist of training deep convolutional neural network models (DCNN) (LeCun et al., 1989) with datasets of urban images that have some sort of label that is used as an estimator for the perception of that urban space, such as Place Pulse 2.0.

However, current deep learning methodologies, have the disadvantage of being ”black boxes”. In other words, they lack a direct or systematic way to explain or interpret the obtained results. This problem comes from the layered structure of the neural network models and from the millions of learnable parameters they contain. Many of the problems in which these models are used would greatly benefit of more human understandable explanations of the results, since it provides more confidence and control over the decisions influenced by the systems, making this a very important area of research for the deep learning field (Adadi & Berrada, 2018; Ras, van Gerven, & Haselager, 2018). For the particular case of urban perception, explainability of the results is of utmost importance, since the added information is valuable for the design of public policy. For instance, it could be use to better discriminate which locations would be better recipients of an intervention, and which elements to modify so it convenes an effective improvement of perception. Despite that, a fully explainable approach is yet to be proposed and the purpose of this work is to progress towards that objective.

The research community has realized the importance of explainability and has taken the research in two main directions: one is to design novel neural network architectures and training methods with the intention of making them interpretable, such as the work by Dong, Su, Zhu, and Zhang (2017). The other direction is to create post-hoc algorithms

(Adadi & Berrada, 2018) that analyze the results given by the neural network. These algorithms commonly use machine learning models, including neural networks (Ghorbani, Wexler, Zou, & Kim, 2019). In particular, the work by Rossetti et al. (2019), presents an approach that uses semantic segmentations of images (Badrinarayanan, Kendall, & Cipolla, 2015) as input for a discrete choice model that estimates an utility function quantifying the perception of citizens regarding different concepts, such as beauty and safety. Among other things, this approach allows for a post-hoc aggregated analysis of the results, based on the coefficients of the utility functions, which quantify the importance of each of the explainable input variables. Is important to note that this type of techniques usually imply a trade-off between the prediction performance of the model and it's explainability.

TODO:ESTO HAY QUE ARREGLARLO AL FINAL

The objective of this work is to design and train a model for the urban perception problem, that can give explainable insights on an instance level. For that it proposes a novel solution, consisting of a neural network architecture, that is end-to-end trainable and by using semantic segmentation (Zhao, Shi, Qi, Wang, & Jia, 2016) and self attention mechanisms (Vaswani et al., 2017) can show explainable insights for each of the input images.

The remainder of this manuscript is organized as follows, Chapter 3 summarizes relevant previous research. In chapter 4 the problem is formally defined and the proposed model is described. Chapter 5 gives details on model implementation and training. Finally, in chapter 6 presents the research results and 7 the final conclusion.

2. RELATED WORK

This chapter consists of two sections, the first one shows an overview some of the different methods that have been previously used in the literature for understanding or quantifying urban perception. The second section summarizes the main aspects of the research on explainability on deep learning, and describes some techniques that have been applied in urban perception or other domains that are relevant for this work.

2.1. Understanding and quantifying urban perception.

2.1.1. Classic approaches.

Methods for measuring perception of urban spaces have been part of the literature of several disciplines for many years, with some of the most influential studies dating back to 1960 (Lynch, 1960). Due to technological limits the literature consisted mainly of different types of qualitative surveys for a long time. These surveys consisted in having subjects complete different tasks such as drawing maps of a certain place (Lynch, 1960), evaluating fundamental aspects of a neighborhood (Nasar, 1990), or in more recent approaches evaluating the impact of transformations generated with edited images (B. Jiang, Mak, Larsen, & Zhong, 2017). Most of these surveys were conducted in person or by phone, and then the results were analyzed manually, making it very difficult and costly to scale to multiple locations, or larger amounts of samples. The main benefit of this approach, is that it allows for a precise control of the observation process since both the subjects being interviewed and the spaces in question are chosen by the researcher. Furthermore, the experiments conducted in person allow for the observer to use senses different than vision to analyze the subject space, resulting in a richer appreciation.

A different methodology, more common in economics and engineering, consists of using discrete choice models and stated choice surveys to model the effect of different variables in perception or other urban related variables (Rose & Bliemer, 2009; Iglesias et al., 2013; Torres, Greene, & Ortúzar, 2013). The amount and complexity of the variables

measured depends on the model design. To have and exact control of the variables that have an effect on the survey, computer generated images of urban spaces can be used (Iglesias et al., 2013; Torres et al., 2013).

The advantage of this method is that through the estimated parameters of the model, the effect of each of the studied variables on the perception estimation can be measured, allowing for quantitative results and an understanding of the impact different elements have on the perception of the urban landscape. The main disadvantage of this approach comes from the difficulty of the survey design, variables need to be chosen carefully and the process its vulnerable to biases from the model designer.

2.1.2. Pure machine learning approaches.

Thanks to the massive adoption of web and mobile technologies such as Google Maps, new types of data are available in considerably large volumes, and new highly scalable ways of generating data can be designed and implemented quickly. These facts allows the application of data-intensive machine learning algorithms to new problems, including urban perception estimation. Several different datasets have been proposed for this problem, most of them based on surveys over large amounts of urban images (Salesses et al., 2013; Dubey et al., 2016; Quercia, O'Hare, & Cramer, 2014; Liu, Silva, Wu, & Wang, 2017; Santani et al., 2018). The most used, all consisting of pairwise comparisons of street view images, are *Place pulse 1.0* (PP 1) (Salesses et al., 2013) with measures of safety, class and uniqueness over images of 4 cities, *Urban Gems* with measures of beauty, quietness and happiness over images of London and *Place pulse 2.0* (PP 2) (Dubey et al., 2016), the largest dataset available, with measures of six different attributes over images of 56 different cities. All of these were collected through public online surveys of large scale, where the users are asked to choose the image most representative of an attribute of a pair, see figure 2.1 for an example.

Earlier attempts at using these data for training models focused on turning the perception quantification into a classification problem by first ranking the images from the votes

Figure 2.1. Snapshot of the place pulse 2.0 survey. Extracted from Dubey et al. (2016)

with manually engineered methods, such as the one suggested in Salesses et al. (2013), and then using the rank to split the data in two halves with a different label. Ordonez and Berg (2014) use this approach to train SVM models on PP 1 using different types of visual features as input, including features generated by a deep neural network (Donahue et al., 2014). On the PP 2 paper, the authors present the first end to end deep learning model for urban perception regression, which uses a typical transfer learning technique (Pan & Yang, 2010), a Imagenet (Deng et al., 2009) pretrained network for the base of the model, which is used as input for by two parallel modules, one for classification and one for regression. They train the architecture separately on the 6 different attributes of the dataset, the models learn to emulate human voting and to output a urban perception score (through the regression module) on the image for the correspondent attribute. Other works (Porzi, Rota Bulò, Lepri, & Ricci, 2015; Santani et al., 2018) take similar approaches but pretrain models or use features based on the places dataset (Zhou, Lapedriza, Xiao, Torralba, & Oliva, 2014), which provides better performance according to their results.

F. Zhang et al. (2018), train models on PP 2 by combining a DCNN features and a SVM classifier, they use this model to obtain perception indicators of Beijing, they also use a semantic segmentation model (Cordts et al., 2016) on the images and used the results as input to a linear regression, interpreting the regression weights as an indication

of importance of the different segmentation classes on perception. On a following work (F. Zhang et al., 2020) they train one deep network to predict all 6 attributes of PP 2 in one forward pass, they do this using an end-to-end architecture similar to Dubey et al. (2016) but adding one output and loss component for each attribute.

Is important to note that most of the literature so far is more focused on applying the models to new cities (F. Zhang et al., 2018; Santani et al., 2018; Costa, Soares, & Marques, 2019; Rossetti et al., 2019) or generating new datasets with new attributes (Santani et al., 2018; F. Zhang et al., 2020), than it is on improving model design and performance. This is consistent with the fact that so far no good measures of performance for this problem have been defined, due to the fact that the datasets don't provide a measure of perception per se but a proxy through the survey votes. The objective of the models in the literature is to rank the images by the estimated perception of an attribute, but they measure performance using accuracy on classification of the human votes, which doesn't necessarily correlate with the models capacity to generalize and rank well, especially in conflicted cases where even human voters would have difficulties (F. Zhang et al., 2018). Despite the fact that models in the literature don't surpass 70% classification accuracy on PP2, the actual ranking task seems to have correct results either by visual inspection, or by comparing with metrics from other domains such as crime rates or wealth indicators (Rossetti et al., 2019; F. Zhang et al., 2018; Ordonez & Berg, 2014).

2.1.3. Mixed approaches.

With the intention of generating more or different insights, usually more explainability, some work in the literature consists of combinations of computer vision or machine learning methods with other techniques. In Rossetti et al. (2019) the authors use a combination of low and high level features of the images as input for a discrete choice model that calculates perception. They extract low level features with traditional computer vision methods like edges or blobs and the high level features with a pretrained neural network

for semantic segmentation. The semantic segmentation features allow for a posthoc analysis of the results, the authors reach conclusions like ”Images with more sidewalks were deemed to be safer, livelier and wealthier, but less beautiful on average” and they present a table with the significance of each of the segmentation classes in each of the six PP 2 attributes according to the discrete model parameters. On a similar line, as was mentioned earlier F. Zhang et al. (2018) in addition to their main method, use semantic segmentation features (they aggregate them by percentage of pixels on the image) as an input for multivariate linear regression allowing for similar conclusions to those of Rossetti et al. (2019) but using the beta coefficients (see figure 2.2).

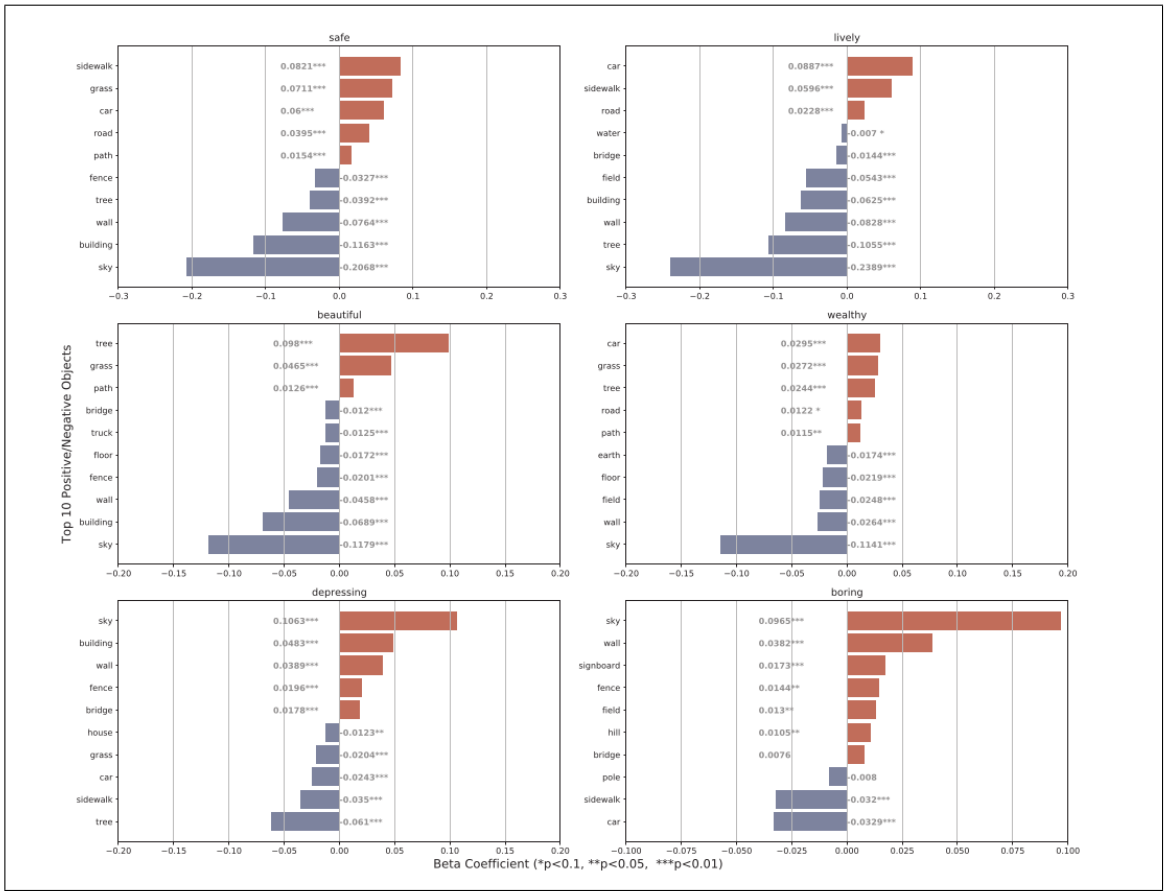

Figure 2.2. Linear regression beta coefficients for most significant objects. Extracted from F. Zhang et al. (2018)

On another work Seresinhe, Preis, and Moat (2017) train a DCNN to calculate the beauty of outdoor images, using transfer learning from the Places dataset, but separately they use a places trained model to obtain text tags from the scenes such as 'Mountain' or 'Tower', and similarly to F. Zhang et al. (2018) they use a regression model (elastic net) to make conclusions about the significance of the concepts on the perception of beauty. The disadvantage of this approaches is that they give more insights of the results only at a general level, and therefore do not allow for conclusions on a per instance level, which is what this work intends to do.

Authors of Costa et al. (2019) do an agreement analysis for this type of datasets, they built their own dataset of pairwise comparisons for safety, but used it for generating clusters of users based on the semantic segmentation of the images they voted for. They conclude that most clusters are due to lack of enough comparisons to do a good characterization and that given enough votes all users converge to one generic profile. Is important to note that authors don't provide any social or demographic information of the 439 users that participated in the survey, and no other similar studies have been done so far so their conclusion hasn't been replicated.

2.2. Explainability in machine learning.

As was mentioned before, explainability has become a very active area of research in machine learning, this is due to the large increase in the usage of ML models for different day to day applications that affect the life's of thousands of people (Ras et al., 2018). For example, in cases where model outputs are used for analytics or decision making, explainability can make the model both more trustworthy and informative.

Adadi and Berrada (2018) summarize the reasons for enhancing explainability in four points:

(i) Explain to justify: To fullfil the need for reasons of a particular ML generated outcome.

(ii) Explain to control: To allow a better handling of model behavior.

(iii) Explain to improve: The additional understanding of model outputs is useful to design improvements on the systems.

(iv) Explain to discover: As a model overcomes human performance in a task, if its doing so in an explainable manner, then new knowledge for humans may be obtainable.

Is also important to note that laws and regulations related to this topic may become norm in the future such as with the *European Union General Data Protection Regulation (GDPR)* (2016). According to it's articles 13, 14 and 15, when personal data is collected for automated decision-making, the subject has the right to access, and the data controller is obliged to provide, "meaningful information about the logic involved as well as the significance and the envisaged consequences of such processing for the data subject", which will be very difficult to comply with, when working with something like a black box neural network.

One of the two most common approaches to explainability in the literature are Post-hoc methods (Adadi & Berrada, 2018) which try to obtain insights about how the models work, after the process of inference over all the dataset is completed. The methods mentioned in section 2.1.3 are examples of this approach. Other more complex methods found in the recent literature are based on analyzing model sensitivity to semantically meaningful concepts on input (Kim et al., 2017; Shi, Zhang, Wang, & Reddy, 2020), concepts that may be automatically mined from the data as in the approach proposed by Ghorbani et al. (2019). This techniques, although very promising, are still too recent and are not extensible to many domains.

The other approach, which is the one followed by this work, consists of taking advantage of the model design to improve interpretability. This can be done by either using existent features of the model or by introducing architectural changes that make them more explainable. A traditional example of this approach are rule based models like decision

trees (Breiman, Friedman, Stone, & Olshen, 1984). Due to the black box nature of deep neural networks this becomes a much more complex task for deep learning, and its an important area of research.

Earlier solutions found in the literature consist of augmenting model input with semantic information, such as text, object bounding boxes or even knowledge bases (Dong et al., 2017; Zhuo, Cheng, Zhang, Wong, & Kankanhalli, 2019; G. Li, Wang, & Zhu, 2019). This methods usually require additional supervision which restricts them to densely annotated datasets such as Visual Genome (Krishna et al., 2016), and the way the neural networks actually use the additional information is not always clear.

Other very common technique in the literature is the use of attention based models (Bahdanau, Cho, & Bengio, 2014), which have layers that consist on using a part of the input (usually called query) to compute a set of weights for the rest of the input (usually called value). The attention weights are usually computed through a linear transformation and a softmax operation on the query, giving them the property of being a probability distribution over the value vector (Cordonnier, Loukas, & Jaggi, 2019), which is used to increase or decrease, parts of the value vector and therefore the layer output. A particular case of attention is self-attention, which means that the same vector is used as both query and input. A common attention architecture in the recent literature is the transformer (Vaswani et al., 2017), which has been widely adopted in both language and vision tasks (Devlin, Chang, Lee, & Toutanova, 2018; Radford et al., 2019; Bello, Zoph, Vaswani, Shlens, & Le, 2019; L. H. Li, Yatskar, Yin, Hsieh, & Chang, 2019; Carion et al., 2020).

Attention models have the additional value that the weights can be used to interpret what the network is doing, providing explainable information about the model's decision process for each data instance (Wiegreffe & Pinter, 2019). Clark, Khandelwal, Levy, and Manning (2019) analyse the attention outputs of the NLP transformer model BERT and show how they correspond well to linguistic notions of syntax, Y. Zhang, Niebles, and Soto (2019) create an explainable VQA model by adding supervised self-attention layers and visualizing their output as heatmaps over the input images (see figure 2.3). Cordonnier

et al. (2019) present a theoretical relationship between between self attention and convolutional layers, and as part of their work they provide an interactive visualization of the attention weights [1]. M. Jiang, Chen, Yang, and Zhao (2020) present a dataset that includes attention labels for images generated by human eye movement, allowing models to learn correct human like attention patterns. Is clear that visualization of attention weights has become a prominent tool for improving model explainability on the recent deep learning literature not only on language but on vision tasks as well (Z. Zhang, Lan, Zeng, Jin, & Chen, 2020; Johnston & Carneiro, 2020; Carion et al., 2020).

Figure 2.3. Visualization of attention weights for the VQA problem. Extracted from Y. Zhang et al. (2019)

[1] Available at `epfml.github.io/attention-cnn/`

3. DATASET

This chapter presents a deep analysis of the dataset used. On section one an overall description and main statistics are shown. Section two consists of an early analysis of the data along with the preprocessing steps taken. Finally section 3 makes an analytical definition of the problem of learning urban perception from this data.

3.1. Description

As was mentioned previously, this work is based on the Place Pulse 2.0 dataset (Dubey et al., 2016). PP 2.0 is a crowdsourced dataset designed for learning urban perception from street view like images. Unlike regular datasets for supervised machine learning, that have labels for each image, Place Pulse consists of pairwise comparisons between images, and the ground truth is a vote representing which of the images is more representative of an attribute (ties are also possible). That structure makes traditional classification / regression approaches inapplicable, but opens the door for pairwise based ranking techniques, that are more suitable to urban perception since a ground truth for how much an image represents an abstract attribute such as ”safety” it's impossible to define.

The dataset consists of approximately 1.2 million pairwise comparisons of 112,000 images from 56 cities, distributed on 6 attributes: wealthy, safety, depressing, boring, lively and beautiful, making it the biggest available dataset for urban perception. The crowdsourcing survey was active for 5 years and it was answered by 81,630 different users. Demographic information about the users was not collected.

One of the main issues with PP 2.0 is that the authors made only the votes available, supplying only the geographical coordinates of the images, so that they can be downloaded from the google street view API, but changes in the API interface and the available images through out the years have made it increasingly difficult and costly to download the original dataset for new researchers.

3.2. Analysis and preprocessing

As a first preprocessing step all noisy images are removed by using a size threshold, since images small in size are mostly google api errors or unintelligible places like dark tunnels.

Is important to note that, unlike most crowdsourced datasets, the authors of PP didn't do any validation on the votes by making it so that the same question was answered by more than one person, and deleting inconsistent votes. 99.59% of the image pairs that appear in the data set have a single vote in a category (see 3.1 for details.), making it impossible to validate them in any way. Even though some research indicates that answers to this surveys aren't affected by user bias or demographics (Salesses et al., 2013; Costa et al., 2019), the noise in the votes is a clear dataset disadvantage. 34% of the pairs that have more than one vote in an attribute show inconsistencies between the votes.

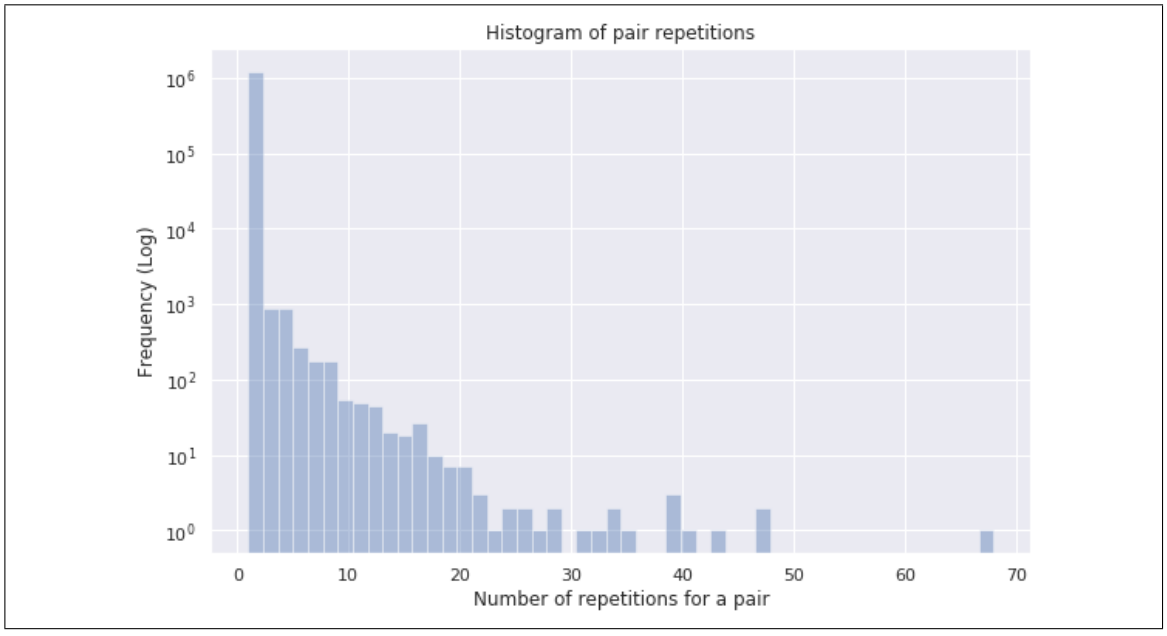

Figure 3.1. Histogram for amount of repetitions for each pair of images

For this work all the inconsistent duplicates are removed and a single vote of those consistent is kept in the dataset. After this steps 1,207,938 votes for 111,299 images are left. See table 3.1 for the exact vote distribution.

Table 3.1. Vote distribution after preprocessing.

Attribute	**# of votes**
Wealthy	150,370
Safety	364,130
Depressing	130,781
Boring	125,744
Lively	263,123
Beautiful	173,790
Total	1,207,938

Users of the survey had the possibility of voting that a pair is tied for an attribute, meaning that they didn't perceive any significant difference, previous works just discard this data and don't use it for learning, focusing only on the votes where a winner was chosen (Dubey et al., 2016; F. Zhang et al., 2018; Ordonez & Berg, 2014). After preprocessing 15.3% of the votes are ties, which means a significant amount of information is lost by disregarding them.

3.3. Problem Definition

Following a similar formulation than Dubey et al. (2016), each attribute A in the PP 2.0 dataset consists of a set of m images $I_A = \{x_i\}_{i=1}^{m} \in \mathbb{R}^{h \times w \times 3}$, with h and w the image height and width respectively, and a set of N vote triplets $P_A = \{(i_k, j_k, y_k) \mid i, j \in \{1, \ldots, m\}, y \in \{1, 0, -1\}\}_{k=1}^{N}$, representing a comparison between the ith and jth image in I with y being the ground truth label, where $y = 1$ or $y = -1$ means a win by image i or j respectively and $y = 0$ denotes a tie.

The objective is to, for each attribute, learn a ranking function $f_A : \mathbb{R}^{h \times w \times 3} \rightarrow \mathbb{R}$ that maps the image tensor to an urban perception score, satisfying the order given by the votes, formally the maximum amount of the following constraints need to be satisfied:

$$y \cdot (f_A(x_i) - f_A(x_j)) > 0 \;\; \forall (i, j, y) \in P_A, \; y \in \{-1, 1\} \tag{3.1}$$

Unlike the previous literature, tie votes are also used in this work, generating the following additional constraints:

$$|f_A(x_i) - f_A(x_j)| < m \;\; \forall (i, j, y) \in P_A, \; y = 0, \; m \in \mathbb{R}^+ \tag{3.2}$$

Where m is a constant margin.

Since f_A is intended to learn a ranking of the input images, it is desirable that the function defines and order on the image space so that the ranking results are consistent. This condition can and should be enforced by model design (Köppel et al., 2019), but since the data is crowdsourced without validation, the constraints generated by equation 3.1, do not represent a 100% transitive order. Because of that, it is infeasible for a model designed for ranking and therefore transitive by construction, to satisfy all of them. This issue makes it harder to obtain high scores in accuracy based metrics in practice, and those are the only ones available in the literature so far.

4. PROPOSED MODEL

This chapter presents a detailed explanation of the neural network models proposed in this work and the correspondent baselines used for comparison. In section one the architectures of the main networks are shown. In section 2 we detailed additional components used on the models and training. Finally, section 3 shows the baselines models used for the ablation study of both performance and explainability.

4.1. Network architectures

As was mentioned before, the main principle followed for model design is to enhance explainability while maintaining performance as much as possible. With that in mind, we combine two state of the art techniques from the deep learning literature, semantic segmentation and attention mechanisms to design three novel architectures that present a significant improvement in explainability over traditional blackbox CNNs. We describe these architectures in the following sub sections, ordered by model complexity. Is important to note that for learning to rank on placepulse 2 forward passes of ranking the network are required for each data instance (one for each image) and both scores are used for calculation of the loss. See section 4.2.1 for details.

4.1.1. SegRank base.

The traditional deep learning approach in computer vision, consists of using a pre-trained CNN (LeCun et al., 1989), on the Imagenet dataset (Deng et al., 2009), such as the ResNet (He, Zhang, Ren, & Sun, 2015), usually called the feature extractor, and then stacking a custom set of layers over its output features. Leaving the CNN weights fixed or updating them on training depends on the particular problem. This is the approach taken by most of the previous literature on urban perception (Dubey et al., 2016; Ordonez & Berg, 2014; F. Zhang et al., 2018).

In this work, we propose replacing the traditional feature extractors for a fully trained semantic segmentation network. The semantic segmentation task consists of assigning a label to every pixel in an image, and therefore it implies a fine grained detection of object edges, providing a rich amount of information that is human understandable. The output of a semantic segmentation model is a probability distribution over the different classes for each pixel, making it usable as a feature map of the image. See figure 4.2 for an example.

We base our models on the PSPNet architecture (Zhao et al., 2016), since it is one of the highest performing models available in the literature. It's design its based on a ResNet50 and a pyramid pooling module, which consists on parallel poolings and convolutions at different scales, that are then concatenated and used to generate the output with a final convolution.

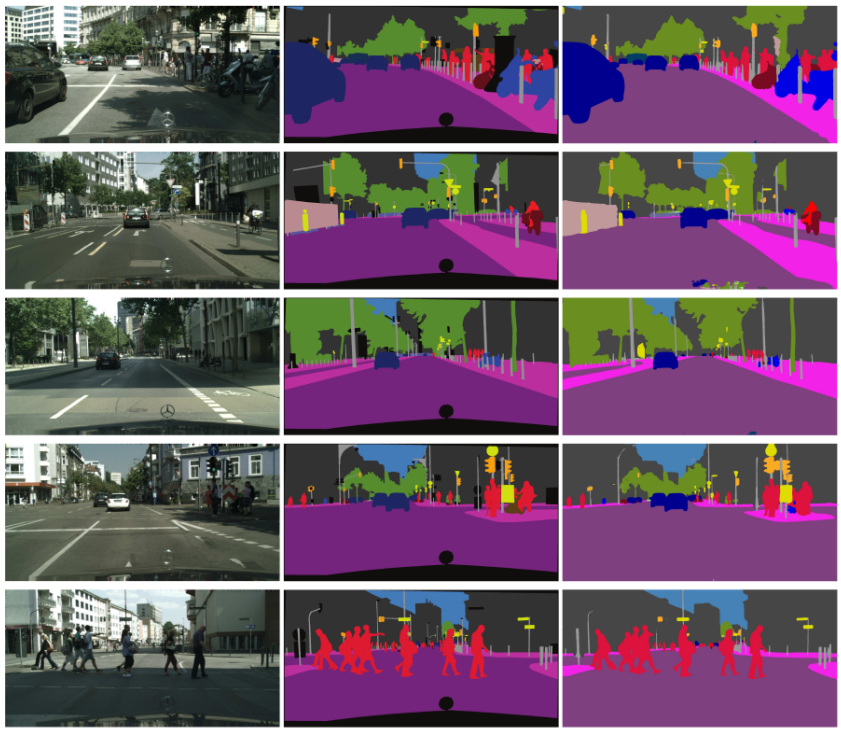

Figure 4.1. Examples of semantic segmentation by the PSPNet model on the CityScapes dataset. Extracted from Zhao et al. (2016)

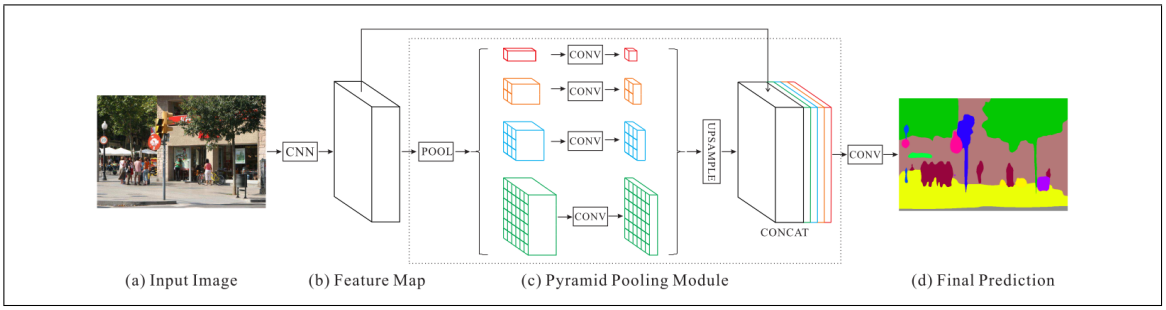

Figure 4.2. PSPNet architecture. Extracted from Zhao et al. (2016)

We train PspNet on the CityScapes dataset (Cordts et al., 2016), since its urban images taken from a car have considerable similarity to street view images, and its classes have proven informative for the urban perception problem in previous research (Rossetti et al., 2019; F. Zhang et al., 2018). After this process we keep the network weights fixed and use the output as features for subsequent layers. The segmentation output is a tensor S, $S \in \mathbb{R}^{h \times w \times C}$, with C the number of different classes. We experiment with using the features directly or applying a softmax operation.

For the calculation of the ranking score, we apply a linear transformation to every pixel distribution, flattening the output to $\mathbb{R}^{h \times w}$ and then an MLP with one hidden layer and ReLU activation. The final linear layer of the MLP generates a single scalar value representing the perception score.

Is important to note that the features given by segmentation are of considerable less dimension than traditional ResNet features making them significantly less expressive. For example, given a standard 244×244 image, ResNet50's *conv_5c* layer outputs a $2,048 \times 8 \times 8 = 131,072$ sized tensor while PSPNet outputs a $19 \times 31 \times 31 = 18,259$ sized tensor. Adding to that, since traditional CNN based approaches allow for finetuning, the amount of trainable parameters is also much smaller for this model than traditional models, and therefore a significant performance drop is likely to happen.

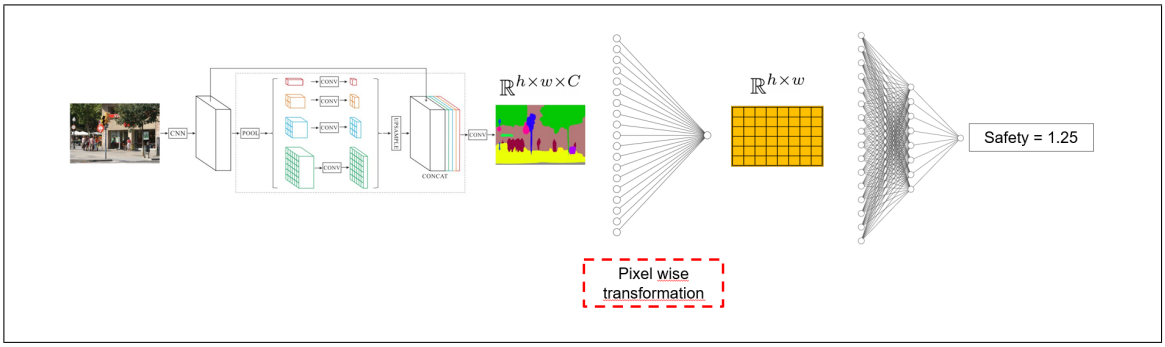

Figure 4.3. First model architecture

4.1.2. SelfSegRank

With the intention of improving performance and explainability of the model, we process the segmentation output with self attention mechanisms instead of a traditional MLP as they have been proven to provide both benefits by previous research (Vaswani et al., 2017; Wiegreffe & Pinter, 2019; Cordonnier et al., 2019).

For our model we use the scaled dot product attention mechanism proposed by Vaswani et al. (2017). We abstain from using the full multi head attention mechanism that consists of the same operations but splitting the input in several ”heads”. We do this because using multiple heads adds complexity to the interpretation of the attention outputs, since different heads may output inconsistent weights, as is mentioned in Clark et al. (2019) and J. Li, Tu, Yang, Lyu, and Zhang (2018) and also verified on this task by our own experiments.

The attention mechanisms receives three matrixes as input: the query Q, the key K and the value V. It calculates a matrix of attention weights over V based on Q and K and the final output is given by the product between V and the weights. A linear transformation is defined for each of the inputs with weights W_Q, W_K, W_V respectively, and another transformation W is applied to the final output. Formally the attention layer can be defined

as:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{4.1}$$

$$\text{AttentionLayer}(Q, K, V) = \text{Attention}(QW_Q, KW_K, VW_V) \cdot W \tag{4.2}$$

Where d_k is the embedding size of the key. For the particular case of self attention, the same input is used as query, key and value, so for our case we make $Q = K = V = S'$ with $S' \in \mathbb{R}^{(hw)\times C}$ and equal to the segmentation output flattened to one spatial dimension.

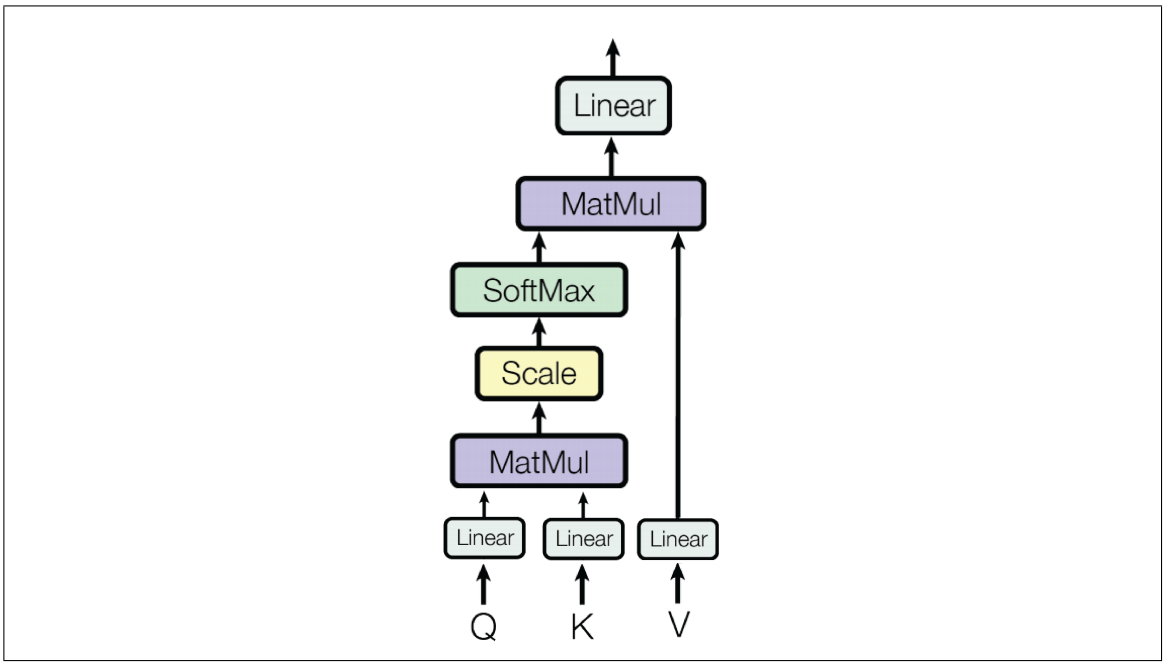

Figure 4.4. Attention layer operations. Adapted from Vaswani et al. (2017)

Similarly to the previous model we apply a linear layer to calculate the ranking score from the attention output. We only use one layer instead of two in this model because the attention mechanism already has a large amount of parameters and a linear transformation of its own.

In parallel the attention weights are also outputted by the model and are used for visualization.

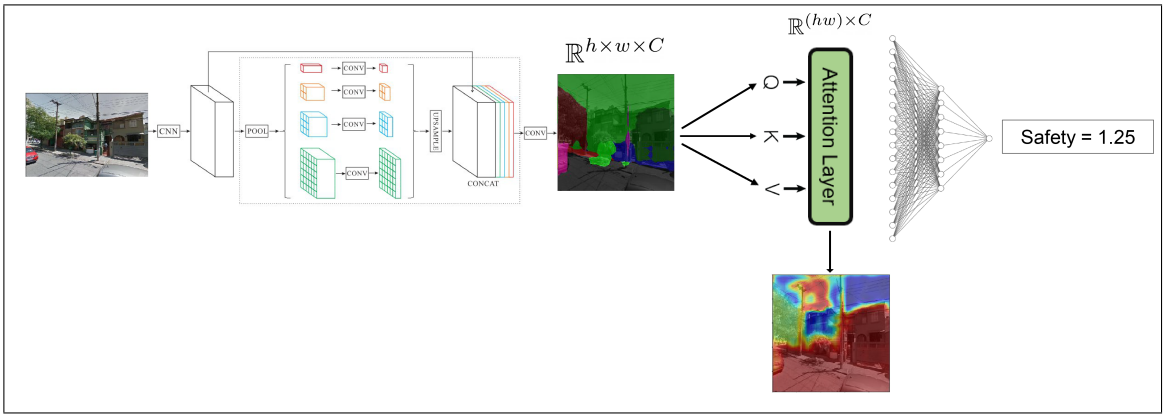

Figure 4.5. Segmentation and self attention network.

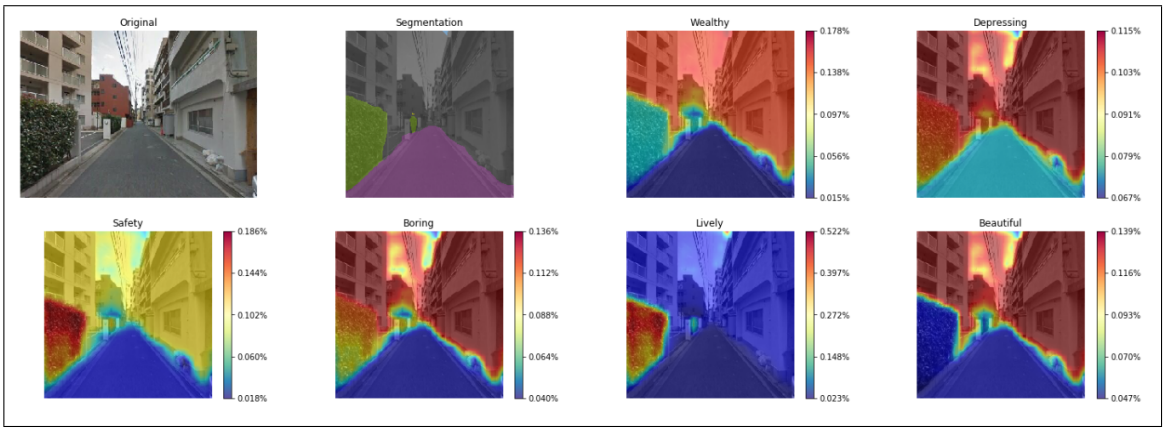

Figure 4.6. Example segmentation and self attention weights for all six attributes.

As it can be seen on figure 4.6, the attention weights keep the object shapes, allowing for a clear interpretation of which objects are significant to the output.

4.1.3. AttentionSegRank

As was mentioned before, segmentation based features are considerably less expressive than traditional deep CNN features and can't be finetuned, generating an important trade off between explainability and model performance. As a solution to that problem, we propose a mixed approach, that weights in both the image segmentation and the CNN features, in order to achieve both good performance and interpretability. To do that, we take advantage of the multiple inputs in the scaled dot product attention mechanism, the

methods consists of using the segmentation as key, and the ResNet features as both query and value (see equations 4.1 and 4.2).

We do this because using the segmentation as key induces the attention weights to maintain a similar shape as the segmentation objects, maintaining interpretability. To understand why this happens, see the QK^t product on equation 4.1 that generates the weight matrix, a single element of the matrix (or a single attention weight) is given by:

$$a_{ij} = \sum_{l=1}^{d} q_{il} k_{lj} \tag{4.3}$$

With a_{ij} being the weight of feature j on output feature i. Setting up $K = S'$ and $Q = F'$, with S' and F' the flattened segmentation and ResNet features respectively:

$$a_{ij} = \sum_{l=1}^{d} f_{il} s_{lj} \tag{4.4}$$

Meaning that the weight of feature j on output feature i depends on which object is j, resulting in an attention weight matrix that keeps the interpretability of the segmentation objects independently of the convolutional features.

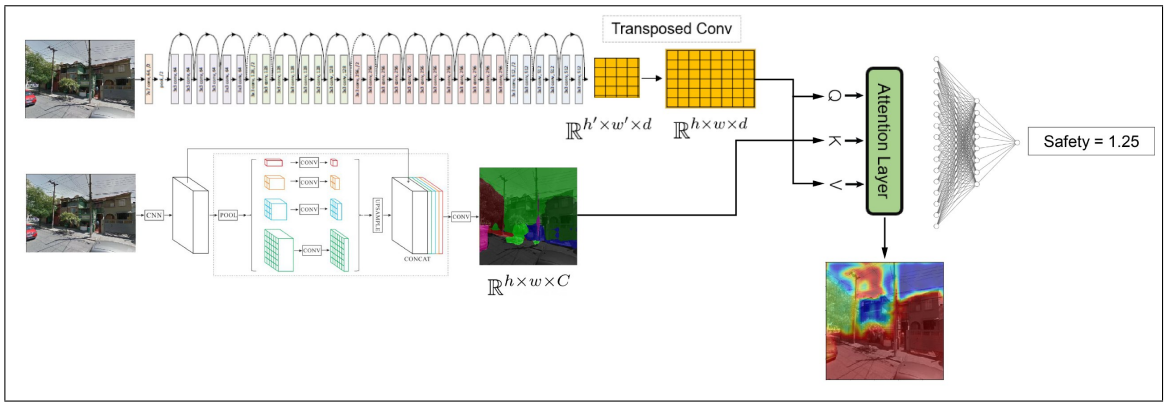

Figure 4.7. Segmentation as key network.

In practice, K and V must have the same spatial dimension for equation 4.1 to be valid, we solve this problem by using layer *conv_4f* of ResNet50 instead of *conv_5c* , since

it has a larger spatial dimensionality, and we add a transposed convolution layer (Noh, Hong, & Han, 2015) to do the final upsampling required to match the segmentation output dimension.

4.2. Additional components

4.2.1. Loss function

The loss function for this task must account for the pairwise structure of the dataset, and should represent the cost of breaking restrictions given by equations 3.1 and 3.2. For 3.1 we use a hinge loss similar to the one proposed by Dubey et al. (2016):

$$L_r(x_i, x_j, y|\Theta) = \max(0, -y(f_\Theta(x_i) - f_\Theta(x_j)) + m_r) \tag{4.5}$$

Where f_Θ and Θ represent the network and its parameters respectively, and m_r is an hyperparameter. This loss component makes it so that the model learns to assign a higher score to the image winner of the vote. Based on the work by Doughty, Damen, and Mayol-Cuevas (2018) we also add a second component so that tied votes can be used for training. According with equation 3.2 we define:

$$L_t(x_i, x_j|\Theta) = \max(0, |f_\Theta(x_i) - f_\Theta(x_j)| - m_t) \tag{4.6}$$

Where m_t is also an hyperparameter. Finally, the complete loss function is defined as:

$$L(x_i, x_j, y|\Theta) = \begin{cases} L_r(x_i, x_j, y) & \text{if } y \in \{-1, 1\} \\ L_t(x_i, x_j) & \text{if } y = 0 \end{cases} \tag{4.7}$$

In practice we take the mean loss over the batch examples and we set $m_r = m_t = 1$

4.2.2. Semantic Dropout

It's important to note, that the semantic segmentation model trained on cityscapes, presents an unavoidable drop in segmentation performance when applied on placepulse due to domain shift. Errors in the segmentation can produce significant problems in the final perception quantification and can also cause confusing attention heatmaps due to errors in the object edges.

In practice we identified a tendency for the models to have attention weights highly biased towards specific segmentation classes, which is highly undesirable both for explainability and model generalization.

We solve these problems by implementing what we call Semantic Dropout, similar to traditional Dropout (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014), but instead of dropping a single neuron with probability p, Semantic Dropout drops the probabilities of an entire segmentation class, inducing artificial errors during training and preventing the network from becoming too sensitive to segmentation errors while also reducing the bias in attention weights. Mathematically this technique is equivalent to the spatial dropout proposed in Tompson, Goroshin, Jain, LeCun, and Bregler (2015), but applied to segmentation probabilities instead to convolutional kernels.

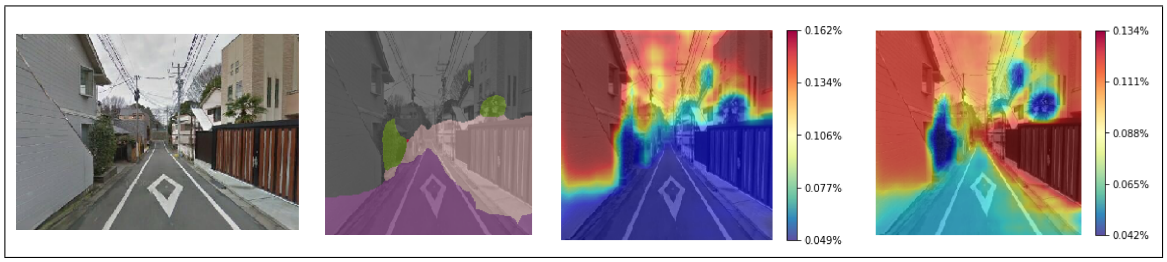

Figure 4.8. Effect of Semantic Dropout. From left to right: original image, semantic segmentation and the outputs of the self attention model trained, without and with semantic dropout. The example shows how the attention weights are less biased towards one class and therefore less affected by the errors in segmentation.

4.3. Baselines

With the purpose of making and ablation study, we also train two baseline models bases on the architecture proposed by Dubey et al. (2016), designed for measuring the effect of the segmentation and attention mechanisms in both performance and explainability of the models. We base these models on the ResNet50 CNN (He et al., 2015), as is the defacto approach for computer vision problems. We abstain from using larger versions of ResNet due to significant overfitting issues.

4.3.1. ResNet50 + MLP

The first baseline consists of a standard finetuned ResNet50 with a two layer MLP. This model doesn't provide any sort of out of the box explainability and therefore is useful to measure how segmentation affects performance. Unlike its segmentation based sibling, dropout and L2 regularization are necessary for training this model, due to the significantly larger amount of trainable parameters that come from finetuning the CNN.

4.3.2. ResNet50 + Self Attention layer + MLP

A baseline on explainability is also important, since improving it is the key contribution of this work. For that we use a similar architecture to attention based explainability models from the literature (Y. Zhang et al., 2019; Cordonnier et al., 2019; Bello et al., 2019), consisting on combining a finetuned CNN with self attention layers.

We take the output of ResNet50's *conv_5c* layers and give it to the attention layer defined in section 4.1.2 and then to a two layer MLP for calculation of the final score.

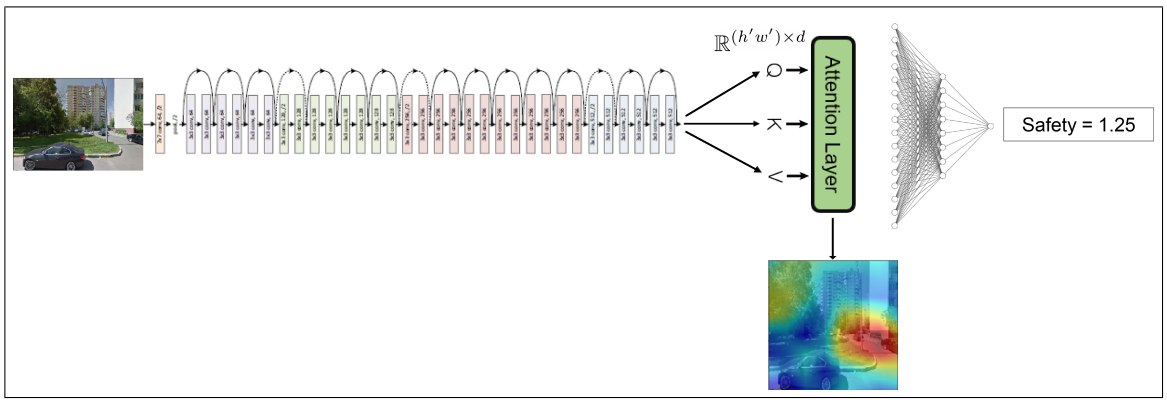

Figure 4.9. ResNet and self attention network. ResNet diagram extracted from He et al. (2015).

5. METHODOLOGY

This chapter shows the practical details of implementation and training.

5.1. Models and training

All of our models are implemented using the Pytorch library (Paszke et al., 2019) version 1.2.0. We use the implementation and pretrained weights of ResNet available on the Torchvision library (Marcel & Rodriguez, 2010). We train our own PSPNet based on the implementation by Huang, Wei, Wang, and Liu (2019). All models are trained using a single 12 Gb Nvidia Geforce-GTX 1080 Ti GPU except for the mixed model, which is trained on a 24 Gb Nvidia Titan RTX.

For training we make a 75%/25% train/validation splits of the dataset for each attribute. We keep the splits fixed for all models, so they all see and are evaluated on the same data. All models are trained for 40 epochs and we keep the model with the best validation accuracy on epoch end.

Table 5.1. Hyper parameters and configurations for each model.

Parameter/Model	**ResNet50**	**ResnetAttn**	**SegRank**	**SelfSegRank**	**AttentionSegRank**
Batch Size	32	32	32	32	32
Learning Rate	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
Opt. Algorithm	SGD	SGD	Adam	Adam	Adam
Finetuning	Yes	Yes	No	No	Yes
Dropout	0.3	0.3	0	0	0.1
Semantic Dropout	N/A	N/A	0	0.1	0.1
Weight Decay	10^{-5}	10^{-5}	0	0	0

Baselines are trained with SGD with a momentum of 0.9 (Rumelhart, Hinton, & Williams, 1986) as it provided better results empirically. For segmentation based models we train with Adam (Kingma & Ba, 2014) and we set ϵ, β_1 and β_2 to 10^{-9}, 0.9 and 0.98 respectively. We use semantic dropout on both models that have segmentation and attention, and add an equivalent regular dropout layer to the ResNetAttn Baseline for fair comparison. Weight decay and traditional dropout are used for all baseline models that finetune ResNet weights. See table 5.1 for details on the training hyperparameters.

5.2. Visualization

For visualization we generate both segmentation and attention images, in this section we will explain how we generate the attention visualizations, for segmentation see appendix A.1.

We generate the attention images by first extracting the weight matrix A from the softmax operation of the attention layer (see equation 4.1). A is a square matrix of size $|A| = (hw)^2$ with h and w the height and width of the layer's input, since it represents the importance of each pixel for each output of the layer. To reduce it to a single weight per pixel we take the column wise (opposed dimension of the softmax) mean of A and then reshape it to the original image size, obtaining an attention map $A' \in \mathbb{R}^{h \times w}_{[0,1]}$.

Since in practice, h and w are too small to produce a good quality visualization we resize A' with bilinear interpolation to the standard size of 244×244. Finally each image is min max normalized to be in the $[0, 255]$ interval in order to apply the color gradient that generates the final heatmap. For sample results see section 6.2.

6. RESULTS

This chapter shows the main results obtained. On section one we present the quantitative performance and training results. Section 2 explains how the different visualizations are generated, including examples for all the models.

6.1. Quantitative results

6.1.1. Model performance

Even though the objective of this research is to learn a ranking (or regression) to quantify the urban perception, exact labels for this are not available, so we have to measure model performance based on the Place Pulse votes, which as was mentioned on section 3.3, has considerable issues. We use as performance measure the equivalent to classification accuracy, considering which image won the vote as the target label. In other words, we evaluate the percentage of restrictions (see 3.1) that are satisfied by the model. We do this separately for each attribute in it's corresponding validation set and the final accuracy value for each model is calculated as the mean accuracy through all attributes.

Both ResNet based baseline models achieve an accuracy of ~66% and as it was expected, replacing the more expressive CNN features for semantic segmentation, caused a significant performance drop, falling to 60.62% for SegRank and 61.43% for SelfSegRank. See table BLABLA, for the exact accuracy values.

6.1.2. Training behavior

Models trained on the place pulse dataset are very prone to overfitting, we believe this is due to it having a very large amount of votes in comparison to the amount of available images, and because the task is very hard to generalize given the high amount of noise that the dataset has from how it was collected. This can be seen clearly on figures 6.1 and 6.2. Both baselines models present considerably overfitting, showing accuracy differences

between seen and unseen data of up to 25%, and ceasing to improve on the validation set after one or two training epochs.

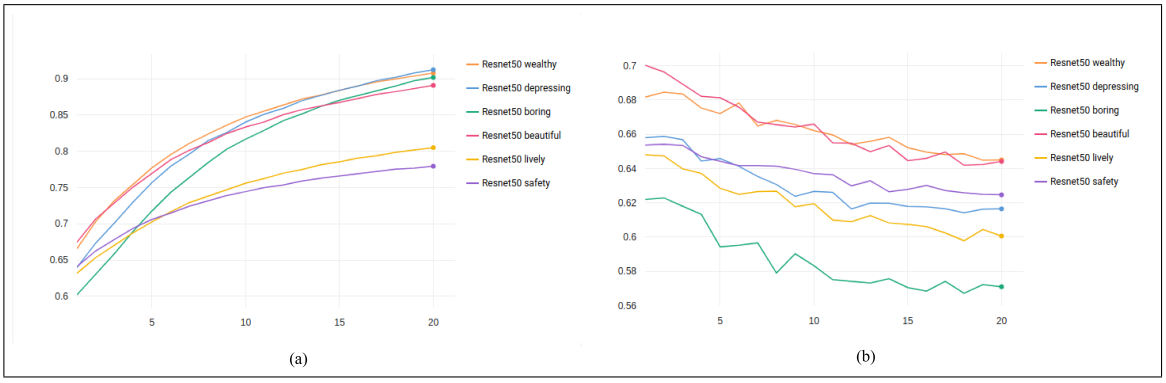

Figure 6.1. ResNet50 baseline accuracy vs epoch learning curves on training (a) and validation (b).

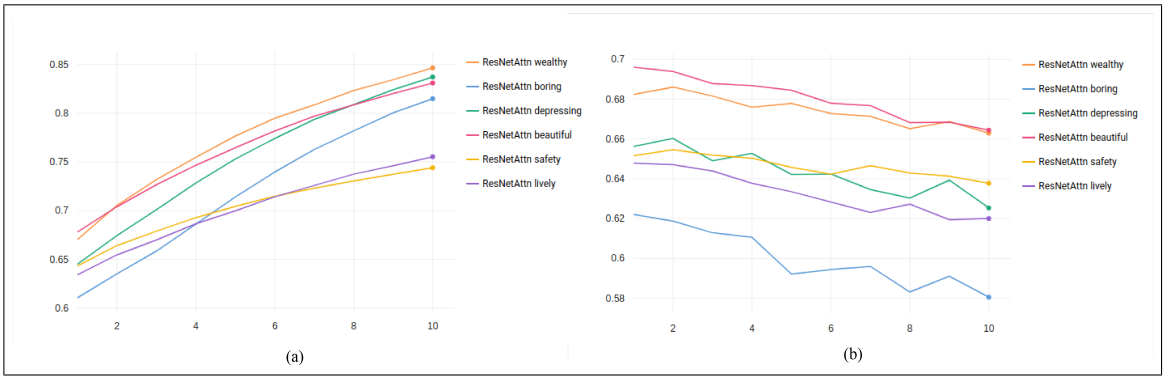

Figure 6.2. ResNetAttn baseline accuracy vs epoch learning curves on training (a) and validation (b).

Replacing the CNN features for semantic segmentation generates a considerable change in training behavior, with the reduced expressiveness of the segmentation acting as a very strong regularizer, overfitting completely disappears, which translates to a drop of around 20% to 30% accuracy in training, but of only 6% on validation.

The basic SegRank architecture still reaches convergence after one or two epochs. Adding the self attention layer makes it slightly slower allowing the model reach a higher validation accuracy.

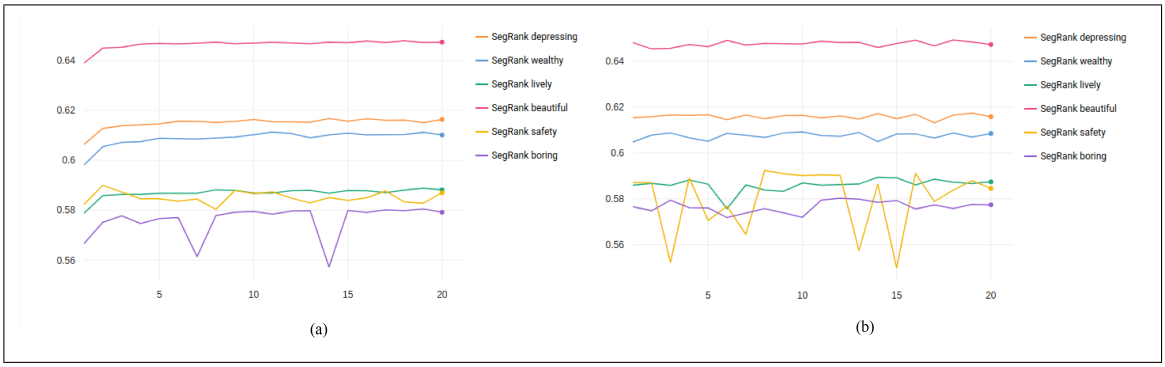

Figure 6.3. SegRank accuracy vs epoch learning curves on training (a) and validation (b).

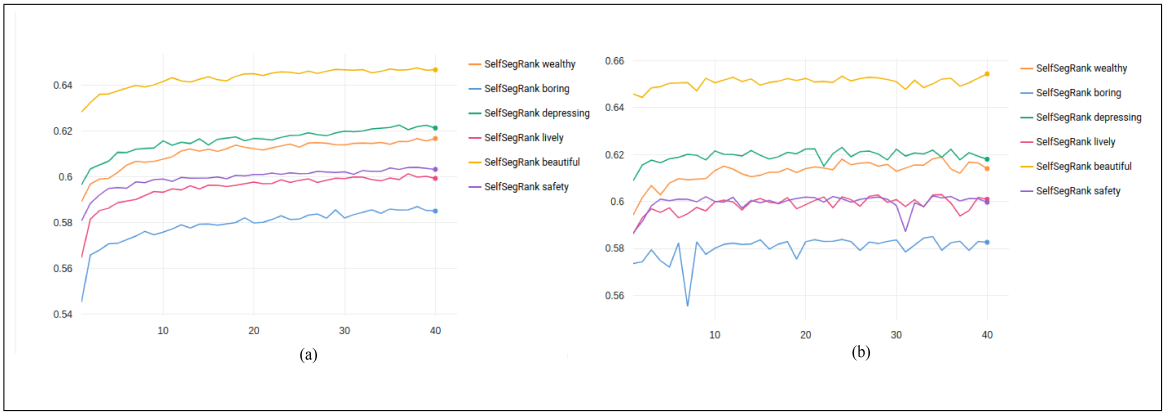

Figure 6.4. SelfSegRank accuracy vs epoch learning curves on training (a) and validation (b).

As it can be seen on the results of all models, the learning process is consistent through out the different attributes. The accuracy of the different attributes is also consistent across the different models, with boring and beautiful being the hardest and easiest tasks to learn respectively on all models.

6.2. Visualization results

- Per attribute visualizations.

- Per object visualizations

- Baseline vs segrank vs segattn

7. DISCUSSION

7.1. Effect of semantic segmentation on learning.

7.2. Relationship between urban perception and semantic segmentation

7.3. Relationship between urban perception and attention

8. CONCLUSIONS

Nothing to say. Be happy.

8.1. Contribution to the state of the art

REFERENCES

Adadi, A., & Berrada, M. (2018, 09). Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE Access*, *PP*, 1-1. doi: 10.1109/ACCESS.2018.2870052

Antonakos, C. L. (1995). Environmental and travel preferences of cyclists. *Transportation Research Part A: Policy and Practice*, *29*(1), 85.

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2015). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *CoRR*, *abs/1511.00561*. Retrieved from `http://arxiv.org/abs/1511.00561`

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bello, I., Zoph, B., Vaswani, A., Shlens, J., & Le, Q. V. (2019). Attention augmented convolutional networks. In *Proceedings of the ieee international conference on computer vision* (pp. 3286–3295).

Breiman, L., Friedman, J., Stone, C. J., & Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A. M., & Zagoruyko, S. (2020). End-to-end object detection with transformers. *ArXiv*, *abs/2005.12872*.

Clark, K., Khandelwal, U., Levy, O., & Manning, C. D. (2019). What does BERT look at? an analysis of bert's attention. *CoRR*, *abs/1906.04341*. Retrieved from `http://arxiv.org/abs/1906.04341`

Clifton, K., & Ewing, R. (2008, 03). Quantitative analysis of urban form: A multidisciplinary review. *Journal of Urbanism: International Research on Placemaking and Urban Sustainability*, *1*, 17-45. doi: 10.1080/17549170801903496

Cordonnier, J.-B., Loukas, A., & Jaggi, M. (2019). On the relationship between self-attention and convolutional layers. *arXiv preprint arXiv:1911.03584*.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., ... Schiele,

B. (2016). The cityscapes dataset for semantic urban scene understanding. *CoRR*, *abs/1604.01685*. Retrieved from http://arxiv.org/abs/1604.01685

Costa, G., Soares, C., & Marques, M. (2019). Finding common image semantics for urban perceived safety based on pairwise comparisons. In *2019 27th european signal processing conference (eusipco)* (pp. 1–5).

Deng, J., Dong, W., Socher, R., Li, L., Kai Li, & Li Fei-Fei. (2009). Imagenet: A large-scale hierarchical image database. , 248-255.

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2018). BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, *abs/1810.04805*. Retrieved from http://arxiv.org/abs/1810.04805

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., & Darrell, T. (2014). Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning* (pp. 647–655).

Dong, Y., Su, H., Zhu, J., & Zhang, B. (2017). Improving interpretability of deep neural networks with semantic information. *CoRR*, *abs/1703.04096*. Retrieved from http://arxiv.org/abs/1703.04096

Doughty, H., Damen, D., & Mayol-Cuevas, W. (2018). Who's better? who's best? pairwise deep ranking for skill determination. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 6057–6066).

Dubey, A., Naik, N., Parikh, D., Raskar, R., & Hidalgo, C. A. (2016). Deep learning the city: Quantifying urban perception at a global scale. *ArXiv*, *abs/1608.01769*.

European union general data protection regulation (gdpr). (2016). Retrieved from https://gdpr-info.eu/

Ghorbani, A., Wexler, J., Zou, J. Y., & Kim, B. (2019). Towards automatic concept-based explanations. In *Advances in neural information processing systems* (pp. 9273–9282).

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. *CoRR*, *abs/1512.03385*. Retrieved from http://arxiv.org/abs/1512.03385

Huang, Z., Wei, Y., Wang, X., & Liu, W. (2019). *A pytorch semantic segmentation toolbox.* https://github.com/speedinghzl/pytorch-segmentation-toolbox.

Iglesias, P., Greene, M., & Ortúzar, J. d. D. (2013). On the perception of safety in low income neighbourhoods: using digital images in a stated choice experiment.

Jiang, B., Mak, C. N. S., Larsen, L., & Zhong, H. (2017). Minimizing the gender difference in perceived safety: Comparing the effects of urban back alley interventions. *Journal of Environmental Psychology*, *51*, 117–131.

Jiang, M., Chen, S., Yang, J., & Zhao, Q. (2020). Fantastic answers and where to find them: Immersive question-directed visual attention. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 2980–2989).

Johnston, A., & Carneiro, G. (2020). Self-supervised monocular trained depth estimation using self-attention and discrete disparity volume. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 4756–4765).

Khisty, C. J. (1994). Evaluation of pedestrian facilities: beyond the level-of-service concept. *Transportation Research Record*, *1438*, 45-50.

Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., & Sayres, R. (2017). Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). *arXiv preprint arXiv:1711.11279*.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Köppel, M., Segner, A., Wagener, M., Pensel, L., Karwath, A., & Kramer, S. (2019). Pairwise learning to rank by neural networks revisited: Reconstruction, theoretical analysis and practical performance. In *Joint european conference on machine learning and knowledge discovery in databases* (pp. 237–252).

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., ... Fei-Fei, L. (2016). Visual genome: Connecting language and vision using crowdsourced dense image annotations.. Retrieved from https://arxiv.org/abs/1602.07332

Laing, R., Davies, A.-M., Miller, D., Conniff, A., Scott, S., & Morrice, J. (2009). The application of visual environmental economics in the study of public preference and urban greenspace. *Environment and Planning B: Planning and Design*, *36*(2), 355-375. doi: 10.1068/b33140

LeCun, Y., Bengio, Y., & Hinton, G. (2015, 05). Deep learning. *Nature*, *521*, 436-44. doi: 10.1038/nature14539

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, *1*(4), 541–551.

Li, G., Wang, X., & Zhu, W. (2019). Perceptual visual reasoning with knowledge propagation. In *Proceedings of the 27th acm international conference on multimedia* (p. 530–538). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3343031.3350922` doi: 10.1145/3343031.3350922

Li, J., Tu, Z., Yang, B., Lyu, M. R., & Zhang, T. (2018). Multi-head attention with disagreement regularization. *arXiv preprint arXiv:1810.10183*.

Li, L. H., Yatskar, M., Yin, D., Hsieh, C.-J., & Chang, K.-W. (2019). Visualbert: A simple and performant baseline for vision and language. *ArXiv*, *abs/1908.03557*.

Liu, L., Silva, E. A., Wu, C., & Wang, H. (2017). A machine learning-based method for the large-scale evaluation of the qualities of the urban environment. *Computers, Environment and Urban Systems*, *65*, 113–125.

Lynch, K. (1960). *The image of the city (vol. 11).* MIT press Cambridge, MA, USA.

Marcel, S., & Rodriguez, Y. (2010). Torchvision the machine-vision package of torch. In *Proceedings of the 18th acm international conference on multimedia* (p. 1485–1488). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/1873951.1874254` doi: 10.1145/1873951.1874254

Nasar, J. L. (1990). The evaluative image of the city. *Journal of the American Planning Association*, *56*(1), 41-53. doi: 10.1080/01944369008975742

Noh, H., Hong, S., & Han, B. (2015, December). Learning deconvolution network for semantic segmentation. In *Proceedings of the ieee international conference on computer vision (iccv).*

Ordonez, V., & Berg, T. L. (2014). Learning high-level judgments of urban perception. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Computer vision – eccv 2014* (pp. 494–510). Cham: Springer International Publishing.

Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345-1359.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems 32* (pp. 8024–8035). Curran Associates, Inc. Retrieved from http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Porzi, L., Rota Bulò, S., Lepri, B., & Ricci, E. (2015). Predicting and understanding urban perception with convolutional neural networks. In *Proceedings of the 23rd acm international conference on multimedia* (pp. 139–148).

Quercia, D., O'Hare, N. K., & Cramer, H. (2014). Aesthetic capital: what makes london look beautiful, quiet, and happy? , 945–955.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, *1*(8), 9.

Ras, G., van Gerven, M., & Haselager, P. (2018). Explanation methods in deep learning: Users, values, concerns and challenges. In *Explainable and interpretable models in computer vision and machine learning* (pp. 19–36). Springer.

Rose, J. M., & Bliemer, M. C. J. (2009). Constructing efficient stated choice experimental designs. *Transport Reviews*, *29*(5), 587-617. Retrieved from https://doi.org/10.1080/01441640902827623 doi: 10.1080/01441640902827623

Rossetti, T., Lobel, H., Rocco, V., & Hurtubia, R. (2019). Explaining subjective perceptions of public spaces as a function of the built environment: A massive data approach. *Landscape and Urban Planning*, *181*, 169-178.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, *323*(6088), 533–536.

Salesses, P., Schechtner, K., & Hidalgo, C. A. (2013, 07). The collaborative image of the city: Mapping the inequality of urban perception. *PLOS ONE*, *8*(7), 1-12. Retrieved from https://doi.org/10.1371/journal.pone.0068400 doi: 10.1371/journal.pone.0068400

Santani, D., Ruiz-Correa, S., & Gatica-Perez, D. (2018). Looking south: Learning urban perception in developing cities. *ACM Transactions on Social Computing*.

Seresinhe, C. I., Preis, T., & Moat, H. S. (2017). Using deep learning to quantify the beauty of outdoor places. *Royal Society open science*, *4*(7), 170170.

Shi, T., Zhang, X., Wang, P., & Reddy, C. K. (2020). A concept-based abstraction-aggregation deep neural network for interpretable document classification. *arXiv*, arXiv–2004.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(56), 1929-1958. Retrieved from http://jmlr.org/papers/v15/srivastava14a.html

Tompson, J., Goroshin, R., Jain, A., LeCun, Y., & Bregler, C. (2015). Efficient object localization using convolutional networks. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 648–656).

Torres, I., Greene, M., & Ortúzar, J. d. D. (2013, 07). Valuation of housing and neighbourhood attributes for city centre location: A case study in santiago. *Habitat International*, *39*, 62–74. doi: 10.1016/j.habitatint.2012.10.007

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *CoRR*, *abs/1706.03762*. Retrieved from http://arxiv.org/abs/1706.03762

Wiegreffe, S., & Pinter, Y. (2019). Attention is not not explanation. *arXiv preprint arXiv:1908.04626*.

Zhang, F., Zhou, B., Liu, L., Liu, Y., Fung, H. H., Lin, H., & Ratti, C. (2018). Measuring human perceptions of a large-scale urban region using machine learning. *Landscape and Urban Planning*, *180*, 148–160.

Zhang, F., Zu, J., Hu, M., Zhu, D., Kang, Y., Gao, S., ... Huang, Z. (2020). Uncovering inconspicuous places using social media check-ins and street view images. *Computers, Environment and Urban Systems*, *81*, 101478.

Zhang, Y., Niebles, J. C., & Soto, A. (2019). Interpretable visual question answering by visual grounding from attention supervision mining. In *2019 ieee winter conference on applications of computer vision (wacv)* (pp. 349–357).

Zhang, Z., Lan, C., Zeng, W., Jin, X., & Chen, Z. (2020). Relation-aware global attention for person re-identification. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 3186–3195).

Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2016). Pyramid scene parsing network. *CoRR*, *abs/1612.01105*. Retrieved from `http://arxiv.org/abs/1612.01105`

Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., & Oliva, A. (2014). Learning deep features for scene recognition using places database. In *Advances in neural information processing systems* (pp. 487–495).

Zhuo, T., Cheng, Z., Zhang, P., Wong, Y., & Kankanhalli, M. (2019). Explainable video action reasoning via prior knowledge and state transitions. In *Proceedings of the 27th acm international conference on multimedia* (p. 521–529). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3343031.3351040` doi: 10.1145/3343031.3351040

APPENDIX

A. SEMANTIC SEGMENTATION

A.1. Visual representation

For visually representing segmentation, we make a color map over the images, following the cityscapes color palette (Cordts et al., 2016). See figure A.1 for the exact palette and class list, and figure A.2

Figure A.1. Segmentation color palette.

Figure A.2. CityScapes sample.