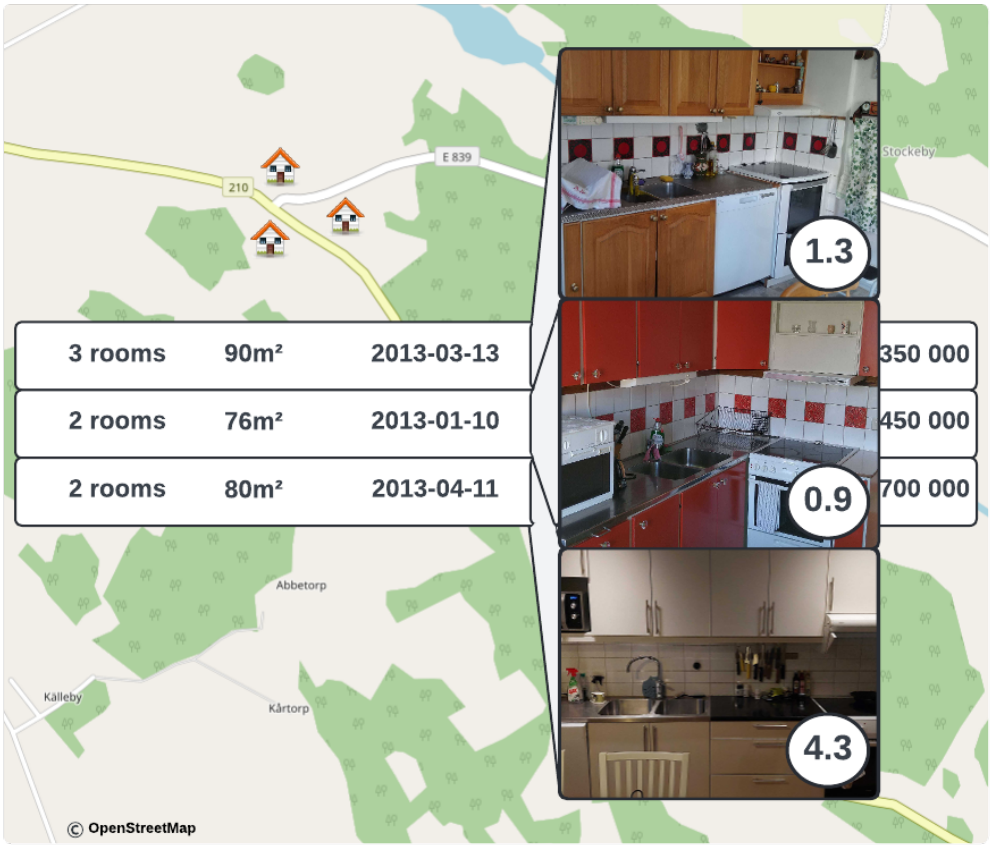
CHALMERS
UNIVERSITY OF TECHNOLOGY



Incorporating Interior Property Images for Predicting Housing Values

Master's thesis in Data Science and AI

Adrian Gortzak

Nedim Can Ulusoy

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2024

Master's thesis 2024

Incorporating Interior Property Images for Predicting Housing Values

Adrian Gortzak

Nedim Can Ulusoy

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
Gothenburg, Sweden 2024

Incorporating Interior Property Images for Predicting Housing Values

Adrian Gortzak
Nedim Can Ulusoy

Supervisor: Milad Malekipirbazari, Computer Science and Engineering
Advisor: David Magnusson, Valueguard Index Sweden AB
Examiner: Aila Särkkä, Mathematical Sciences

Master's Thesis 2024
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Visual features as part of the comparative market analysis tool.

Gothenburg, Sweden 2024

Incorporating Interior Property Images for Predicting Housing Values

Adrian Gortzak
Nedim Can Ulusoy
Department of Computer Science and Engineering
Chalmers University of Technology

Abstract

The property valuation process for the real estate market is essential for predicting a fair market value. This process is traditionally carried out by brokers, including inspecting and assessing the subject property to find comparable sales for comparative market analysis (CMA). Meanwhile, an automated valuation model (AVM) can help achieve an autonomous version of this process, which speeds up the process but lacks some of the inputs that a manual assessment provides. AVMs have difficulty considering more subjective architectural qualities, such as beauty, stability, and utility, due to the difficulty of quantifying these aspects objectively. New advancements in Visual Transformers (ViT), self-supervised learning and Contrastive Language-Image Pre-training (CLIP) technologies have shown favourable improvements in the field of computer vision. Therefore, this study explores the potential improvements of these new techniques within the visual feature extraction task to enhance the AVMs from interior images. By applying ViTs as binary classifiers, clusters, and textual descriptions matching, we aim to enrich the feature extraction process for a property valuation model in the region of Uppsala County, Sweden. Our findings show modest enhancements in the AVM's performance, which align with prior studies, but also highlight that these new technologies can extract more detailed features compared to previous methods. Furthermore, they demonstrate the potential for these technologies to capture more comprehensible architectural qualities from images, which could significantly assist brokers in the valuation process.

Keywords: Computer Vision, Transformers, Feature Extraction, Machine Learning, Deep Learning, Real Estate, Automated Valuation Models, Architectural Qualities.

Acknowledgements

Firstly, we want to extend our sincere gratitude to Milad Malekipirbazari, our academic supervisor, for quickly providing suggestions and answers to our inquiries. Additionally, he suggested alternatives and supplied a solid foundation in the field of AI and ML while still being patient with us.

Secondly, we would also like to show appreciation and gratitude to Valueguard Index Sweden AB is for an exciting research area, hardware access, and an educative process. Moreover, we would like to thank David Magnusson, our company supervisor, for his engagement, fast support and industry expertise in guiding the thesis forward and resolving issues along the way.

Finally, we want to express our heartfelt gratitude to all the individuals who have contributed feedback and input throughout the thesis.

Adrian Gortzak & Nedim Can Ulusoy , Gothenburg, 2024-06-17

I want to express my heartfelt appreciation to my partner, Sandra, for her invaluable help and emotional support throughout the thesis. I am deeply grateful for her encouragement and support.

Adrian Gortzak, Gothenburg, 2024-06-17

I would like to express my heartfelt gratitude to my family for their unwavering support and encouragement throughout the entirety of my academic journey.

Nedim Can Ulusoy, Gothenburg, 2024-06-17

Contents

List of Figures

List of Tables

1

Introduction

This section introduces our research topic of visual feature extraction in real estate, outlines the research question, and provides a brief background on automated valuation models and visual feature extraction with deep neural networks.

1.1 Background

Property value assessment is an essential part of the real estate field, ensuring that both buyer and seller get a fair price for what is typically one of the most significant investments in their lifetime. The aim of property value assessment is to predict *market value*, which is the expected value of a property under normal conditions [1]. A broker conventionally does this with the help of a manual *Comparative Market Analysis (CMA)*, which uses comparable sales to derive the market value [1].

While thorough, the traditional method can be time-consuming, especially when comparing *architectural qualities* that require an assessment of *utility*, *stability*, and *beauty*. Quantifying these aspects automatically poses a challenge, often requiring a manual visual comparison of the images.

To address this challenge, the real estate field has seen significant advancements in *Automated Valuation Models (AVMs)*, which efficiently and accurately estimate a property's market value efficiently and precisely [2]. While AVMs also primarily rely on easily quantifiable data, such as the living area and the number of bedrooms, computer vision and pattern recognition improvements have made it feasible to combine unstructured data, such as images, as part of the AVMs' input.

Previous studies have tried incorporating visual features from images [3], [4]. They have focused on exterior and interior images, using techniques such as *Convolutional Neural Network (CNN)* [5] for pattern recognition related to market value. While these studies have shown feasibility, they have also shown only modest improvements [3].

Building on this foundation, new computer vision improvements have shown promising results in quantifying aesthetics and outperforming the earlier state-of-the-art CNN models [6] with the recent development of *Vision Transformers (ViT)* [7].

Furthermore, self-supervised methods, such as *Self-Distillation with No Labels (DINO)* [8] and *A Simple Framework for Contrastive Learning of Visual Representations*

(SimCLR) [9], have demonstrated the ability to learn robust features from images without relying on labelled data. They achieve this by utilising parts or transformed version of the image during training of the ViT model, predicting whether the images originate from the same source. These methods reduce the previous need for a larger labelled image dataset to create robust models for feature extraction.

Additionally, the arrival of *Contrastive Language-Image Pre-Training (CLIP)* [10], trained to find the similarities between the ViT model and a text encoder, makes scoring the images by textual input feasible without additional training or provided examples.

1.2 Problem

This thesis aims to solve the problem of extracting and incorporating visual features related to architectural qualities and utilising advancements in computer vision, particularly via ViTs to reduce the uncertainty in AVMs. Through methods such as classification, clustering, self-supervised learning, and CLIP, this research aims to extract more comprehensive information from interior images, surpassing the limitations of previous techniques. These improvements could potentially revolutionise both automated and manual valuation processes.

Another problem with similar studies is the impact of cultural differences and preferences when focusing on different countries. The preferences for particular architectural styles and access to material or functionalities of rooms may cause significant shifts in data distribution. For instance, due to differing market dynamics and European trends, the pattern found in the United States study [11] may not be directly applicable or effective in the Swedish housing market.

Therefore, this thesis is an exploratory study to quantify visual features from listing images using *Deep Neural Networks (DNNs)*, such as CNN and ViT, to improve AVMs in Sweden. These findings aim to find the answer to the research question:

RQ: *Which visual features predict market value most significantly in Sweden?*

1.3 Outcomes

This project aims to create an extraction system similar to the flowchart in Figure 1.1. It first takes images as input and proceeds to classify the type of room in these images. The final step is to extract features from these images that are relevant to the architectural qualities, using models tailored to the specific room type identified and utilising deep learning techniques, such as CNNs and ViTs, in the extraction process. These extracted features aim to be used as inputs for the AVMs, allowing for the comparison with AVMs lacking visual features. In addition to assessing changes in accuracy, the importance of the features will be extracted from the model and compared to show the impact of the extracted features.

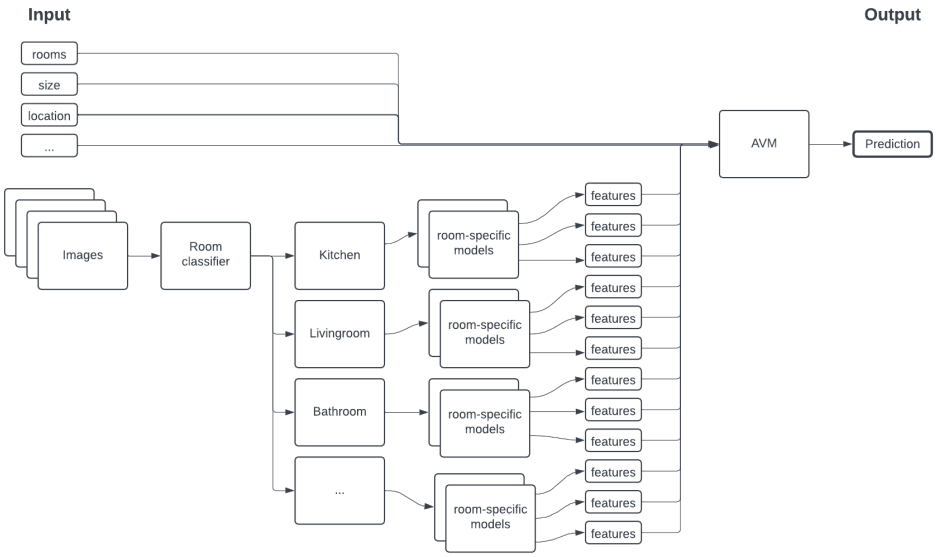


Figure 1.1: Blueprint for the system's process flow

1.4 Structure of the Thesis

The thesis structure will begin with a theoretical summary of the field of housing valuation and previous attempts to use images to enhance the performance of AVMs. The study also explores the visual features connected to market value in Sweden, their importance, and the visual features used in similar studies in other regions. This thesis aims to highlight the most promising techniques and visual features to focus on, based on their proven impact on housing valuation, to answer our research question.

Given this deeper understanding of the Swedish housing market and previous research on visual extraction, the methodology section describes the methodology used within this thesis and the necessary pre-processing to focus on different room types individually. Additionally, it seeks to describe the different automated valuation models and the visual feature extraction techniques applied.

Following the description of the methodology employed in the thesis, the results section highlights the data used in sales and images. Additionally, it discusses the features identified from the different models in this study and their correlation with market value. The thesis ends with a conclusion section, where the results from the experiments are discussed, and further directions are recommended for further exploration.

2

Theory

This section introduces the field of housing valuation, current manual and automated methods to estimate market value, and the potential use of visual data to enhance these models. Additionally, previous research utilising images and computer vision within the valuation process will be highlighted.

2.1 Property Valuation

The property valuation process holds significant importance across numerous fields. In the private sector, property purchases are among the most expensive decisions an individual makes in their lifetime [12]. Ensuring a fair price is essential for both the seller and buyer during negotiations and transactions [13].

Unlike other major purchases, such as a car, a property serves not only as a utility but also as an investment with a potential resale value that historically tends to increase over time [14], [15]. This dual nature of property as both a necessity and an investment underscores its significance as future capital appreciation becomes a key consideration [16]. Furthermore, property valuation plays a crucial role in risk assessments [17], and is utilised by banks during loan assessment [18].

2.1.1 Market Value in Real Estate

In the real estate field, *market value* refers to the most probable selling price in a fair and open market without personal relations between the seller and buyer or coercion and with enough marketing time [1]. This implies equal access to information and opportunities for all parties involved in the selling process. It also implies that the broker would have enough preparation time, and it also requires the seller to be patient without urgency in the selling process. Additionally, the lack of a personal relationship between the broker, the buyer and the seller ensures that the price is not dependent on the personal relation, thereby preventing potentially biased pricing.

Therefore, the sales can be seen as samples from the distribution where the market value is the mean and can not be observed. The sales price is thereby used as the target, with the aim of estimating the market value, assuming that all the sales follow the market value conditions.

Consequently, a slight difference in the market value estimation and the selling price

is expected. However, a significant difference between the two may indicate either a bad estimation or a sale price that does not follow the conditions [1].

2.1.2 Comparative Market Analysis

In Sweden, the broker usually performs a manual appraisal of houses and apartments using a comparative market analysis (CMA) [1]. This process determines the estimated market value of the *subject property* by finding similar comparable property sales, referred to as *comps*, in an area that also shares similar characteristics.

The sale date of the comps must also be close to the valuation date to be comparable due to market trends and price changes over time [1]. Alternatively, if it is necessary to use an older sale, a housing index, such as the one derived from the *Hedonic Regression Model* [19], can be used to track changes over larger areas and take the feature of the property under control to isolate changes over time. Thereafter, the index can be used to adjust the expected market value of the valuation date as either the current or the past date, depending on the specific valuation requirements.

To derive the estimated market value, one can either use the average sale price per square meter of the comps or the *Purchase Price Coefficient* (K/T) value, which is the sale price divided by the taxation value for the comps. Depending on the chosen sub-methods, this is then multiplied by the subject property's corresponding living area or taxation value [1].

Suppose a significant distinction between the comps and the subject property; for example, a worse condition or another addition makes them different. In such a situation, the broker can make additional adjustments to align the price with the predicted market value [1].

Predicting the market value accurately is generally difficult [20], considering the real estate market's very competitive nature and frequent price fluctuations [16].

2.1.3 Automated Valuation Models

The AVM is an automated way of estimating the market value without doing a deeper and more time-consuming analysis manually, especially when large numbers of properties need to be assessed. In addition, it provides an automated early indicator for potential sellers before a broker does a deeper analysis. Multiple types of models can be used to do this process autonomously.

Firstly, a straightforward AVM approach imitates the traditional manual evaluation process using a comparable approach. This approach entails a k-nearest neighbour-like search for comps that can then be used to estimate the subject property automatically [21].

Secondly, linear regression methods are also widely used, especially as an improvement baseline [22]. These methods find the linear relationships between the features and the target, assuming linearity in relations.

Thirdly, Tree-based methods and gradient-boosted methods, such as the *eXtreme Gradient Boosting (XGBoost)* [23], work well when there are no linear relationships and have shown promising results on the AVM task in previous studies [2].

In recent years, the use of Neural Networks has also shown promising results, especially in combining different sources of data into one model. Notably, market leader Zillow transitioned from a multi-model approach to a larger Deep Neural Network (DNN) model, resulting in improved performance and reduced maintenance [24].

2.2 Features Impacting Property Price

Understanding the importance of finding similar comps is crucial for a reliable price estimation. These comps should closely mirror the factors of subject properties. In the *Swedish market value process* [1], Fredrik Brunes categorises these factors into two main groups: those related to the property and the property's location.

Apartments and houses have additional separate features that connect to the market value, such as housing cooperatives for the apartment and land area for the house. Therefore, they are handled separately in the models and the theory [1].

2.2.1 Architectural Quality

The features connected to the property and location are scored on *architectural quality*, drawing from the foundational principles in *the 10 Books on Architecture* by Marcus Vitruvius Pollio [25]. These criteria are divided into three main parts when scoring the architecture: *stability*, *utility*, and *beauty*, which are commonly regarded as *standard* in Sweden. However, due to the unclear meaning and different interpretations of standard, this thesis will use a specific definition of architectural quality [1].

2.2.2 Factors Connected to Market Value in the Location

Starting with location, which is assessed similarly for both houses and apartments, the utility of the location can be evaluated by considering the proximity to positive factors.These can be the distance to marketplaces, commuting possibilities, and workplaces. Additionally, it entails factors such as a sense of safety and relaxation and the availability of places for socialisation with friends and family, such as parks or forest areas within walking distance of the property [1].

Secondly, the area's beauty can be assessed based on the quality of the surrounding houses, streets, and parking spaces in terms of material and detail perspective. This assessment includes factors such as the amount of daylight the area receives, whether views are long or blocked by high buildings and accessibility of the area including multiple routes to access the property [1].

Finally, the stability of the location is assessed based on the materials used in the nearby houses and common areas, such as parks and streets. This includes evaluating if these areas are well maintained and stating any damages and their severity [1].

The location is assessed in multiple layers, which include micro-location, the surrounding area, and the neighbourhood's reputation. The micro-location is the region with a direct connection to the property. Additionally, the surrounding area is the region within walking distance. Lastly, the neighbourhood's reputation is a wider region where the general opinion is assessed. In particular, the reputation is not assessed based on architectural quality but rather as an independent value [1].

2.2.3 Factors Connected to Market Value in the Property

Unlike the nearby area, which is shared between multiple residences, a property offers a private space that the owner can customise. This gives the owner more control over the home environment and architectural qualities.

Although there are distinctions between houses and apartments, there are also several commonalities in terms of the layout, utility, and aesthetics of the rooms. This section explains the overlapping characteristics of the apartment and the house, while the differences will be addressed in the subsequent section.

Stability within the property is connected to the material and building techniques employed during its construction. This relates to the predicted maintenance requirements in the form of expected repairs and the associated cost. Typically, the foundation has longer intervals between repairs than the flooring and walls, but it comes with a comparably high repair cost at the time of repair [1], [26].

The property's utility relates to how it can be used, and this relates to the ability to spend time with friends, cook food, maintain hygiene, and recover through sleep. This could be in the form of a larger room, making it possible to spend time together, sound isolation that keeps noise out, or a bathroom or laundry machine within the residence [1], [26].

Within the property, beauty is related to finer details in pleasing materials and openness in combination with the balance of natural light. It also includes a balance between open and closed areas and the generality of the home, as well as, the ability to use rooms for multiple purposes [1], [26]. While aesthetic preferences may vary, certain features are generally considered appealing, while others, such as damaged walls or broken details, are not.

2.2.4 Property Type Specific Factors

Given the explanation of the common characteristics, the focus now shifts to the differences. Swedish apartments are usually part of a housing cooperative. A monthly fee determined by the financial status and planned maintenance is paid to the cooperative. High fees add to the buyer's expenses, particularly if the cooperative has high loans and may need to increase fees during times of high loan rates. Thus, understanding the cooperative's financial situation is crucial, enabling buyers to assess future potential costs [1].

On the other hand, the house also comes with extensions, such as land, the possibility of extra buildings, and a foundation that is part of the residence. Unlike apartments,

where utilities are shared responsibilities within the cooperative, houses typically place these responsibilities on the owner. This increases the potential repair cost and underscores the importance of assessing the comp's current state during the selection process [1].

2.2.5 Other Factors Connected to Price

There are other factors that affect the subject property's price, such as demand and supply, regulation changes, mortgage rates, and disposable income rates. Since the comps are supposed to be sold close in time to the subject property or adjusted by indices to be, these can therefore be assumed to be shared between the comps and the subject property.

2.3 Images in the Sales Process

When it comes to property sales, images play a crucial role as part of the listing material, providing potential buyers with a first impression of the property. They give the buyer a general idea of the property and assess if they are interested in bidding or attending a viewing [13].

Capturing people's interest is essential in persuading potential buyers to pursue further steps, increasing the number of potential buyers and, thereby, the demand for the property. This part of the selling process has generated a niche in the company area of home staging. It takes advantage of the importance of aesthetics, aiming to make the home feel and look better to attract more buyers. Consequentially, it potentially increases the price and thereby makes the service a potential investment [4].

In a study using eye tracking, it was determined that the subjects spent 60% of the time watching the images in the property advertisement compared to the description and comments from the broker [27]. This highlights the importance of the images in the sale process. In addition, images have the advantage of universally communicating the property's condition without language barriers, conveying its appearance in a way that words alone may struggle to achieve [16].

2.4 Interior Visible Features

A broker will visually inspect the subject property during the valuation process to identify the previously mentioned features related to the property's beauty, utility, and stability, which are necessary to find suitable comps [1]. The property's rooms show material choices related to stability. They can also indicate the feasibility of spending time with friends and family or whether this area is too cramped. Simultaneously, the bathroom and kitchen conditions can indicate one's ability to cook food and maintain proper hygiene [1], [26]. Damages and aesthetics are also visible components on the surface layer of the exterior and interior. These damages can be moisture and humidity on the ceiling or walls, as well as cracks and stains.

Within the home, both in the exterior and the interior, there are time-typical features that are normal for the era [28], [29]. These features can be the type of wallpaper, the doors and the windows of the property or some additional details. While these features can show a desirable style, they also hint at potential underlying issues; the construction industry has tried different techniques and materials throughout the years, only exhibiting the usual problems long after the construction [28], [30].

2.5 Limitations and Challenges

A limitation in the manual assessment process is that while the subject property can be visually inspected and assessed thoroughly, the comps are usually not readily available for inspection, making it hard to adjust the assessment based on these features [1]. This is especially true when the interior parts are involved, while the exterior and surroundings can be viewed with satellite or street view images.

This can lead to a situation as seen in Figure 2.1, where two similar-looking sales differ in price, making it problematic to assess the features setting them apart and adjusting accordingly.

Figure 2.1: Highlighting comp selection problem

A broker with area knowledge and previous sales experience within that area might have a good understanding of the differences, including the general condition and how to adjust the price accordingly. Conversely, a new broker might be more restricted [1]. The visual aspect, if available, can then aid the broker, leading to a better understanding of the differences, as Figures 2.2 and 2.3 show an excessive example of the potential differences in stability, utility, and beauty

Figure 2.2: Example comp (Desired)

Figure 2.3: Example comp (Undesired)

One of the main limitations of these data-based valuation systems, like the AVM, is their difficulty in grasping unstructured data that has proven important to the market value or data hard to access. These, for example, can be natural light or sound levels from the surrounding area and views from the property [16]. While it is easy to add quantitative data, using unstructured data such as ad descriptions, satellite images, and exterior and interior images of a property, requires more advanced feature extraction techniques. However, extracted features from the unstructured data could provide essential information for comparing comps with the subject property, whether for AVMs or the broker.

These architectural qualities for the home have been referred to as the property's unmeasurable values [26], and the literature on what is considered high within these topics is limited in Sweden [31]. This has triggered new research within the field to find objective guidelines [32], thereby highlighting the difficulty and importance of the topic.

2.6 Review of Similar Studies

Along with advancements in *Machine Learning (ML)* and *Artificial Intelligence (AI)* and the increasing ability to utilise unstructured data, methods based on DNNs have

begun to be used to obtain more objective and data-driven real estate valuations. These methods have also shown promising development in predicting these hard-to-quantify features related to architectural qualities.

The literature contains various studies on real estate valuation and the use of visual features from images, mainly aiming to reduce uncertainty in the assessment. The studies have focused on different properties with overlapping themes, such as attractiveness [11], [33], aesthetics [34], material usage [35], luxury levels [4], the impact of damages [36], and the effects of furniture and unfurnitured images [37].

Most of these studies have been conducted on exterior images, such as satellite images [38], [39], street views [20], [40], possibly due to the ease of accessing this data afterwards, with services such as *Google Street View* [41] and *Google Maps Static* [42]. However, the interior images have also been focused on in some studies, where the room types were assessed separately [43]. Closely related, a study was conducted on the number and location of photos taken from Facebook and how that indicated something beautiful or photo-worthy within the area [44].

These studies have been conducted in many countries, such as China [40], Italy [38], England [39], United States [3], and South Korea [20], highlighting region-specific insights. For instance, research in Beijing, China, showed a negative correlation between water bodies and market value due to pollution [40].

2.6.1 Methods

These studies use different methods to measure visual features and their importance. One method of gathering ratings on aesthetics and damages has been used to catch subjective opinions [37], [43]. In these studies, participants grade the visual features in a comparable fashion, and multiple options are presented. A rating is set on the targeted features, such as damages or aesthetics, and an average rating serves as the objective truth during the training of the models. This has also been done to quantify beauty with the help of natural language processing (NLP), where comments on images are gathered to extract assertions in the form of undesired and desired comments regarding image aesthetics [6].

Another method is to use the error of the original estimate as indicators of the visual features' effect on the price [3]. A negative difference in an area can indicate that the visual aspects found differ from the region negatively. Using multiple examples, the model can find these commonly negative and positive patterns as features that can be added to the assessment. This method is heavily based on the assumption that the estimate's error is based on the visual aspects.

2.6.2 Previous Attempts

In case of the studies that utilised ratings, one of the studies evaluates the effect of features in property images on real estate, and a group of experts uses structured methodologies to evaluate the functionality and aesthetics of furniture [37]. The results showed that this approach was effective in aligning furniture design with

consumer preferences and quality standards.

In the study, "Image-Based Appraisal for Real Estate Using Mask R-CNN" [36], they labelled each image with multiple annotations related to the room conditions, including damages and their severity. This study focused on the lack of importance of the property's current situation based on its image in real estate valuation. In this study, the Mask R-CNN [45] approach published by Facebook AI Research was used, and both defect and damage detection were performed by object segmentation on the interior and exterior images of real estate. The primary purpose was to understand the effect of the defects and damages in the interior and exterior images on the price.

The price error was used as an indicator for undesired visual features in the "House Price Estimation from Visual and Textual Features" [3] study. Specifically, a binary classifier for Curb Appeal of houses was developed. It was based on the error of the previous prediction in combination with *Principal Component Analysis (PCA)* of the Pre-trained ResNet features was developed. The choice of a binary classifier for a good and a bad curb is a simplification over a regression task where the actual difference is the target. While their attempts led to an improvement, it was stated that it was only a modest improvement.

In the context of AVM models, these studies have mainly used models such as *Ordinary Least Squares (OLS)* [46] and XGBoost as baseline models, comparing the result with and without visual features extracted from images. Another study used *recurrent neural networks (RNNs)* [47] to process data from random walks based on the location of properties to embed locality to improve property pricing [16]. During performance evaluations of these models, they generally used *Mean Square Error* (MSE), *Mean Absolute Percentage Error* (MAPE), and R^2. In addition, the results of these studies showed moderate improvements, suggesting that visual features reduce uncertainty and emphasise the need for further research [3], [20].

2.7 Computer Vision

The field of computer vision, which retrieves information from images and video, has been active for a long time. CNN's early breakthrough was its ability to capture patterns, making it possible to classify text and scenery from visual media [5].

2.7.1 Convolutional Neural Networks

Yann LeCun and his collaborators first entered this field in 1998. In the paper "Gradient-Based Learning Applied to Document Recognition" [5], they introduced the use of CNNs for document recognition. After this introduction, they pioneered CNNs' architecture and training methods and showed their effectiveness for two-dimensional shapes, such as handwritten characters.

After LeNet, Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton introduced AlexNet in 2012 in the paper "ImageNet Classification with Deep Convolutional Neural Networks" [48]. AlexNet was deeper than LeNet and used ReLU activation functions to increase the model's performance and GPUs for computation. It also achieved

a top-five error rate of 15.3% on the ImageNet challenge, which was a significant performance among the existing models.

Following this, Karen Simonyan and Andrew Zisserman introduced the VGG networks in 2014, in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition" [49]. VGG's architecture was deeper than AlexNet because of the use of very small 3x3 convolutional filters. The VGG model with this architecture achieved better performance on the ImageNet challenge than other models like, AlexNet. This result showed that using deeper networks with smaller convolutional filters can increase the performance of the model and provide better accuracy in image-oriented tasks.

Subsequently, the ResNet model was introduced by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun in 2016 in the paper "Deep Residual Learning for Image Recognition" [50]. This paper presented a new approach to the problem of vanishing gradients during the training of deeper networks. By incorporating skip connections between layers, ResNet effectively mitigated this issue and achieved superior results across various image-oriented tasks, surpassing the performance of previous architectures such as VGG.

While these larger models are trained with millions of images [51] to find robust features, they can start from a pre-trained stage, where the model has already learned base patterns that can be fine-tuned to a set objective. These larger models must be trained on data that are connected to the target domain at hand. For instance, a large model that is trained on a dataset linked to animals might not have found the patterns that would be useful in real estate. The continuation of openly available datasets, from the *MIT indoor 67* [52], with 15,620 images, to *Places 365* [53], with approximately 8 million training images, gives the ability to train these larger DNN models, which would have been impractical due to resource, time and image constraints.

The AlexNet [48], VGG [49], and ResNet [50] models have shown promising performance in the papers related to classifying scenery and distinguishing between different types of settings, such as cafeterias and classrooms. While not perfect, the models still show room for improvement, especially in the top-1 prediction accuracy, which means the accuracy of the class with the highest probability. That is currently between 50% and 60%. This is still impressive given the 365 options. Demonstrating the ability and slight improvement between the generations of models. Therefore, showing the ability and slight improvement between the generations of models [53].

2.7.2 Transformers

The recent success of transformers in NLP, starting from the paper "Attention is All You Need" [54], started an AI leap with the advancements of new tools such as Bert [55] and GPT [56] models. The significant improvement with transformers is the ability to handle sequential data without having to process them sequentially, a limitation that previous models, such as the RNN, had [54], [55]. It does this with the help of *attention* or *self-attention* mechanism, which is used to set the importance

or weight of part of the inputs during training, thereby keeping attention on the high-weighted parts [54]. This has shown improvements over previous methods [6].

The use of transformers has also entered the computer vision field in the form of *Vision Transformer* (ViT) [57], which is a transformer model designed to push the limits of transformers outside their primary field, NLP, and perform in the computer vision or Image Analysis field. In ViT, the main idea is to let the model learn image structures independently by representing all image inputs as sequences of patches using the attention mechanism of the transformers [57].

The main advantage of ViT over CNNs is that ViT uses these attention mechanisms that are not constrained by the spatial structures on which CNNs are based. This allows the model to focus on the most relevant parts of the input image [57]. This can lead to more efficient processing, especially for tasks that benefit from understanding the global context of the image [57].

2.8 Gaps in the Research

Given the rapid advancements in AI, the vast amount of unstructured data within the real estate field in the form of text, images, and videos, this research area holds significant potential [58]. Specifically, it can leverage these unstructured data to improve the market value assessment.

Also, due to the difficulty of estimating market value, it makes property valuation a good test for new developments in the inclusion of extracted features that are difficult to quantify to reduce uncertainty. Given the current difficulty in quantifying the architectural quality features of the property and its usage in the comp selection, we believe that the future of real estate research around the valuation process will continue to find ways to incorporate more complex components. These components can then be correlated with the market valuation to highlights their importance. Additionally, they can be used to make the selection of comps easier for real estate brokers.

Many features related to the architectural qualities could be extracted, such as sound levels and natural light levels. However, they are also hard to quantify accurately and clearly. Furthermore, acquiring relevant data for these features presents a challenge [21]. As AI models continue to advance and more open data becomes available, the field of property assessment will undergo renewed exploration, exploring the uncertainty and understanding of the correlated features. This includes the dimensions of the architectural quality.

2.9 Future Directions

While the use of CNN and DNNs in the valuation process has been studied, mainly in other regions and with both exterior and interior images, the use of ViT remains limited. Regional differences might also include regional biases in the studies, potentially limiting the scope of the findings to other regions, such as Sweden.

Another area for improvement with these findings lies in the ability to interpret the result. While these methods have shown slight improvements in automated valuation, they are usually hard to use outside of AVMs. A more target extraction, with a visual understanding, could also aid the manual assessment process, helping the broker in the comp selection process and speeding up their workflow.

Therefore, there is a need for future studies of visual features in regions such as Sweden that continue to explore ways of incorporating additional features into the valuation process. This could enhance its accuracy, efficiency, and understanding of the market value and its relation to characteristics as architectural qualities.

3

Methods

The method chapter begins with a summary of the research plan, followed by the data and pre-processing required for this thesis. Thereafter, the theory and best practices for training DNNs are explained. The chapter concludes with the visual extraction methods, the AVM models and the scoring methodology used in this research.

3.1 Research Plan

The primary objective of this thesis is to leverage visual aspects from interior images with the help of state-of-the-art computer vision models to reduce uncertainty in Swedish property valuation. The hypothesis is that interior images reflect the architectural qualities that advanced computer vision models can extract and use in the predictions, thereby improving the AVMs' accuracy.

We test this hypothesis by conducting an empirical research study on private housing within the Uppsala county region in collaboration with *Valueguard Index Sweden AB (Valueguard)* [59].

The collaboration with Valueguard enables us to access listing images taken at the time of sale, along with the metadata associated with the property, selling date, and selling price. It also supplies us with housing indices that track price changes over time, which enables us to adjust these sales to the same date, resulting in a more comparable dataset. Lastly, their extensive expertise in the real estate field provided invaluable guidance throughout the project.

This thesis focuses on the interior images and excludes exterior and surrounding images. These interior images are categorised separately according to room types for a more tailored comparison. One initial limitation in this study is that these labels are not provided, thus requiring extensive manual pre-processing through image classification to obtain the required dataset.

In this thesis, the accuracy of the AVM model is used to measure the impact of these extracted features, measuring the reduced uncertainty in the form of MAPE in the AVM prediction. Thereafter, a 10-fold cross-validation is combined with paired t-tests to test for statistical significance of the added visual features. Additionally, when a statistically significant improvement in the reduction of MAPE is found, the feature weight in the model is examined to understand the importance of the

newly added features. Consequentially, this research plan aims to provide a solid foundation for the research.

3.2 Limitations and Scope

The first limitation in scope is the types of properties explored. This thesis focuses exclusively on apartments and smaller houses, essentially year-round private housing. This approach defers the inclusion of images from interior commercial establishments and summer cottages to future studies. This decision ensures a targeted approach, considering the propertys' distinct customer groups and usages. There is also a limited availability of relevant data for commercial housing for this thesis.

Secondly, this thesis only explores the interior images and excludes images of the property's exterior and those depicting the surrounding area. This, in combination with the focus on the room types separately, creates a high reliance on representative data connected to all the room types for each sale.

Thirdly, this thesis does not collect votes or labels from the broker to use as a target for architectural quality. Instead, it attempts to find these structures in the data. This scope is set to explore the advancement of new AI tools for finding patterns, primarily due to the scarcity of available experts in the field to assist in this process.

Finally, the region's size is limited to control the number of sales and images processed within the study. Choosing a region rather than sampling from the country as a whole is chosen to capture comparable sales in the regions. Therefore, it is decided that only sales within the region of Uppsala County are included, as shown in Figure 3.1. The region is chosen based on the available data, with a preference for regions the authors are familiar with. Furthermore, the region is also considered sufficiently large to generate a dataset big enough to make the use of DNN models meaningful.

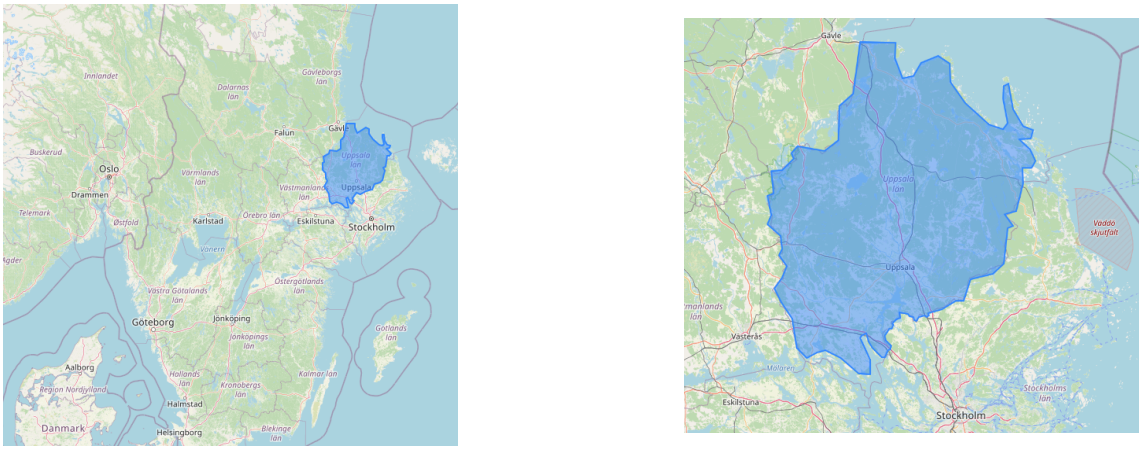


Figure 3.1: Uppsala County on OpenStreetMap [60]

3.3 Technologies

Multiple programming languages can be utilised for visual feature extraction. However, due to previous knowledge and experience, the choice is made to work in Python [61] and use PyTorch [62], as the main library for working with images. Additionally, we use GeoPandas [63] to mark up areas and calculate distances, while we use pandas [64] to load and process the metadata. Subsequently, we run these tools and models mainly in Jupyter notebooks [65], which run inside a Docker container [66] with *graphics processing unit (GPU)* access.

A key tool in the project is *Label Studio* [67], which makes it fast and efficient to work with labeling and ensures that data policies are upheld by working locally. Additionally, it supports the required multi-label and single-class classification that we use in the thesis. The tool is essential for labelling a large quantity of images in a secure and reasonable time with an easy import function from a JSON format to generate the task and hotkeys to speed up the labelling process.

MlFlow [68] is another valuable tool for this thesis. It makes saving experiment results with the connected parameters, scores, models, and graphs more concisely and easily manageable. This reduces the associated difficulties with a more extensive set of models with different training parameters. It also makes it possible to do larger experiments sequentially over multiple days, trying out a wider range of parameters that can be assessed afterwards.

Computing power is an essential part of running these models. Given that DNNs run considerably faster on GPUs than *central processing units (CPUs)* [69]. GPU resources are used to increase the number of feasible experiments within the limited time. Also, due to the sensitive nature of some of this data, an additional requirement is that it has to stay within the company's hardware. This requirement removes the alternative of utilising cloud computing, which can scale more freely. As a result, for this project, a Linux server with an Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz CPU, 32GB DDR4 RAM and an Nvidia GeForce RTX 3090 with 24GB VRAM is provided and used throughout the project. Regarding the dataset and models explored, the hardware is deemed sufficient to run and train the models within a reasonable time for multiple attempts throughout the project.

3.4 Data

Data is our project's most critical and essential part, especially when using the DNN approach due to its data-hungry nature [70]. This data consisted of information related to the property, such as the number of rooms and size. Valueguard provides most of this data through metadata and images connected to the sales. However, some additional data sources are used to enrich the dataset. This additional data is mainly related to the locality in the form of additional regions marked up with the provided coordinates.

3.5 Valueguard Index Sweden AB

Throughout the thesis, there is a collaboration with the company Valueguard. This company has a long history of creating *hedonic housing indices* and aiding brokers in the valuation process by providing CMA tools that suggest comps [71]. Additionally, they offer AVM services in the form of an API [72]. Furthermore, Valueguard gathers its data from various sources, such as realtors, and direct data transfers from multiple real estate agencies as well as large providers such as Svensk Mäklarstatistik [73].

3.5.1 Ethical Considerations

Working with images from people's residences might be intrusive ethically. Although these images have been used for marketing and public viewing, within the thesis, there is a commitment to maintaining confidentiality throughout the project. This is done by implementing strict safety measures to anonymise all images, ensuring that individual privacy is protected and no personal data is compromised. This includes keeping the images secure on the server, generating *universally unique identifiers* (UUIDs) for the images that can only be tied to sales with credentials during the run, and removing any images with individuals from the dataset.

These security measures also led to the decision that the images in the thesis in the form of visual examples are from license free image providers as *Unsplash* [74] rather than the images that are provided.

3.5.2 Metadata

The *metadata* here refers to the data connected to the sale and includes numerical and categorical variables tied to the property and the connected location. The provided metadata is used as the baseline features in the AVM model comparisons.

3.5.2.1 Location

As previously mentioned, the locality is an essential component of the valuation process. Using publicly available regions extended the given base data, giving the models more opportunities to learn characteristics tied to the region.

In this thesis, two external location-oriented sources are used. The first is *Statistics Sweden* [75], which exposes multiple definitions of Swedish regions to capture and compare localities. The regions used from Statistics Sweden in our study include *Demographic Statistical Areas (DeSo)* [76], a geographical division of Sweden for a more detailed demographic and statistical data analysis. Additionally, *Regional Statistical Areas (RegSO)* [77] are used, where Sweden is divided into 3,363 statistical areas based on municipal and county boundaries. *Urban areas* [78] where at least 200 inhabitants are in a contiguous built-up area are also considered. Finally, the study incorporated *Municipalities* [79] that divide Sweden into 290 more extensive regions.

The different regions within the Uppsala County boundaries can be observed in

Figure 3.2 showing the different sizes of the regions and their overlap, allowing the model to tie smaller regions to more extensive regions. Additionally, this is added to ensure the models can capture the micro-location and regional features explored in theory.

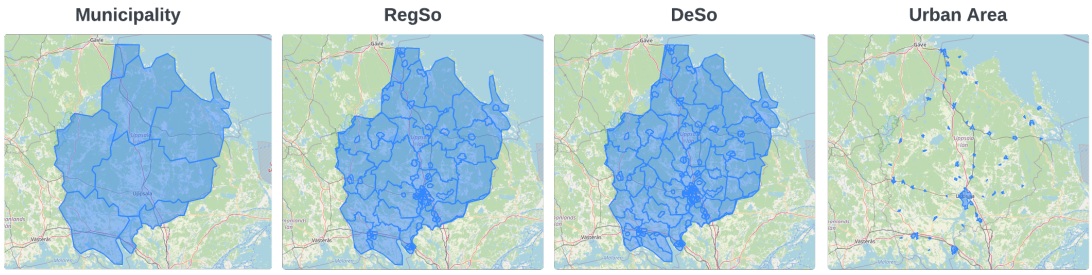


Figure 3.2: Geographical areas from Statistics Sweden on OpenStreetMap

Besides the pre-defined regions provided by Statistics Sweden, *geographical tiling*, similar to a grid system, is used to capture regional characteristics on multiple levels or, as they refer to it, *resolutions*. This is achieved by marking the sales with multiple layers of *Ubers H3 index (H3)* [80] with multiple resolutions. This approach is inspired by a blog post from Zillow about the design of their DNN AVM [24]. Figure 3.3 displays a visual example of the resolution of 7, 9 and 11 in Uppsala centre.

Figure 3.3: H3 Index with different resolutions on OpenStreetMap

One final feature is added to the sale regarding the location, which is the distance to the urban area centre. It aims to capture how far the sales are from desirable locations in the region centre, such as stores and other necessities. The centre is predicted using the 90th quantile on the recounted sale price within the region, thereby only selecting the most expensive sales. The centre point is then generated by taking the median of the x-axis and y-axis of the RT90 coordinates separately. This method operates under the assumption that property market values generally increase the closer they are to the city center. The straight-line distance is then calculated to the center point from the sale within the region and added to the base features.

This study exclusively used the distance to the generated centre of the urban areas. However, other distances, such as the municipality centre, travel distance to the airport, and public transport, can enrich the models with more information regarding the location. Nonetheless, they are excluded to minimise the scope and workload.

3.5.2.2 Base Features and Target

This section describes the base features and the target of the AVM model. Due to the skewed distribution of some of the features, a logarithmic transformation or log scaling is applied to convert these distributions to a more normally distributed one, potentially making it easier for the model to work with these features [81].

The base features in the form of metadata can be seen in Table 3.1 for houses and Table 3.2 for apartments. The type of variable and transformation are indicated with the tag notation of *C* for a *categorical feature*, *L* for a feature with *logarithmic transformation*, and *N* for a *numerical feature*.

Table 3.1: Housing metadata

Feature
Home type [C]
Standard points [N]
Plot area [N]
Living area [N,L]
Rooms [C]
Ancillary area [N]
Construction year [N]
RT90x [N]
RT90y [N]
DeSO [C]
RegSO [C]
Municipality [C]
H3 Index res 3-9 [C]
LKF [C]
Distance to urban area center [N]

Table 3.2: Apartment metadata

Feature
Elevator [C]
Monthly fee [N]
Living area [N,L]
Rooms [C]
Construction year [N]
Floor [N]
Floors [N]
Housing cooperative [C]
RT90x [N]
RT90y [N]
DeSO [C]
RegSO [C]
Municipality [C]
H3 Index res 3-9 [C]
LKF [C]
Distance to urban area center [N]

Regarding the shared features between the property types, *living area* indicates the livable area size. Meanwhile, the *rooms* are the number of rooms within the property and are decided to be categorical due to the shown separation and differences in the number of rooms and the price per square meter [82]. Additionally, the *construction year* refers to when the property is built.

Furthermore, the regional features, *RT90x* denote the horizontal distance to the east from the central meridian of the RT90. Meanwhile, *RT90y* indicates the vertical distance to the north from the central meridian of the RT90. The *DeSO*, *RegSo*, *municipality*, and *H3 Index* with different resolutions are the regions in which the

data is marked up, while the *LKF* represented a region connected to a parish region [83]. Lastly, *distance to urban area center* is the shortest distance to the generated urban area center.

In the case of house-specific variables the *home types* indicate the type of house, such as chain house, semi-detached house or terraced house. Meanwhile, *standard points* reflect the condition scoring of different features in the house, such as kitchen setup and renovations [84]. Additionally, the *plot area* and *ancillary area* indicate secondary areas that are not part of the main living areas of the property, such as the garage or basement.

In the case of apartment-specific features, the *elevator* feature indicates whether the apartment building has an elevator. The *monthly fee* is the monthly rental fee for the apartment to the housing cooperative. The *floor* represents the floor where the apartment is located, while *floors* represents the total number apartment building floors. Meanwhile, the *housing cooperative* is the legal institution that owns the real estate.

The target variable for the AVM, is a logarithmic transformed recounted sale price that is used as a proxy for the market value. In the recount process, the price at the time of the sale is adjusted with the help of a regional index to represent the market value on January 15, 2024. The recounted date is selected because it is the latest published index value at the start of this thesis. This approach allows us to exclude the time variable in our thesis and simplify the task. However, it shall be stated that the further away the original estimate is, the more uncertain the recounted value becomes. In our study, the sales and the corresponding images analyzed cover the period from 2019 to 2024.

The images play a crucial role in our study. Each sale is provided with an average of approximately 30 images related to the property. The room-type images are grouped between the property types. This grouping is based on the assumption that the characteristics of these rooms are comparable and overlap between apartments and houses.

3.6 Room Types

In this study, the focus is primarily on the interior images and handling the room types separately. This is done to make comparisons within the model more meaningful, not comparing kitchens with bathrooms but rather bathrooms with another bathroom to find similarities and differences relevant to the room type regarding the architectural qualities.

Furthermore, the room types used in this study include the bathroom, bedroom, kitchen, living room, and dining room. The room types are decided partly due to the architectural quality connection in the theory, the functional importance of the bedroom for sleep, the importance of the kitchen for making food and the function of handling the hygiene in the bathroom. However, some of the room types, such as dining room are chosen because they were used in similar studies [4]. The goal

of the pre-processing is to obtain images that only depicted the chosen room types, as seen in Figure 3.4.

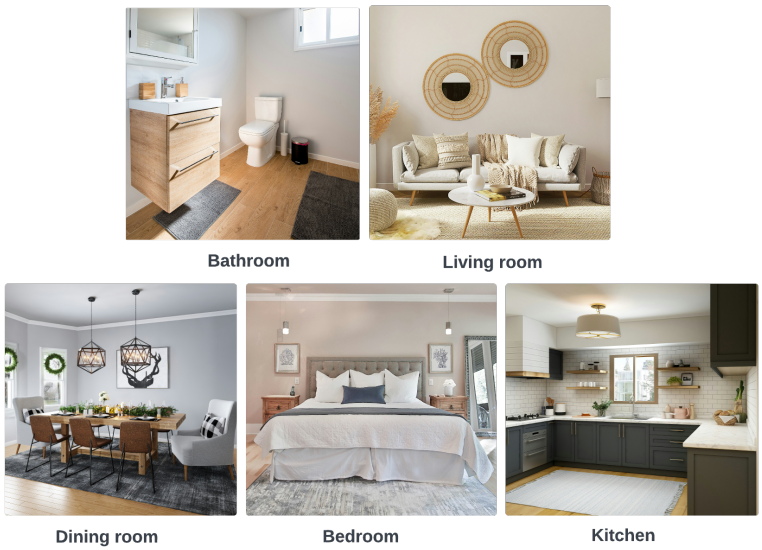


Figure 3.4: Room types explored

3.7 Labelling

Labelling is a time-consuming but essential part of this study. It generated the core dataset by categorising the images to the corresponding room type. A hasty execution of this step can create issues for the rest of the study through a low-quality dataset or a lack of data. Therefore, it is determined that a longer period for labelling will be designated to reduce the impact of time pressure. Each day, a portion of the dataset will be labelled. The data quality is also reviewed throughout the thesis to maintain its high quality. This is done by excluding low-quality and undesirable images, and marking them when seen.

Due to the previous success in distinguishing scenes with high accuracy [4], [53], the labelling task is split into two stages, using a so-called hierarchical approach with two sub-tasks. The first stage aimed to exclude a larger proportion of the irrelevant images from this study, thereby maximising the study related images for a more thorough review in a second stage. The classes of the initial stage included interior, exterior, and others, where others are a class for floor plans and 3D renderings of the property. These three classes are selected due to their apparent visual differences and the assumption with previous successes that they will be easily split with high accuracy.

The second stage, which involved classifying the room types from the labelled interior images, presented a more complex challenge. This is partly due to the presence of multiple room types in one image. This issue is handled with the help of a multi-label classifier, where each room type is represented as an individual probability

vector for each image. Thereby, adding the option to select multiple rooms in the second labeling stage.

In the study, 10,000 images are labeled in the first stage and 5,000 in the second stage. The higher number in the first stage is due to the ease of labelling them, and an early accurate model gives more interior images to the second stage and thereby increasing the number of images that can be used and reducing additional filtering out non-interior images.

Furthermore, the first stage is considered a single-classification task where the highest-scoring class is chosen. In contrast, the second stage is considered a multi-label task where each class is assessed separately with a threshold. This threshold is determined by testing a range of thresholds and selecting the one with the highest F1 score of each room type to ensure a balance between the models precision and recall. A visual explanation of the output of the two stages can be observed in the visual representation of the label processes seen in Figure 3.7.

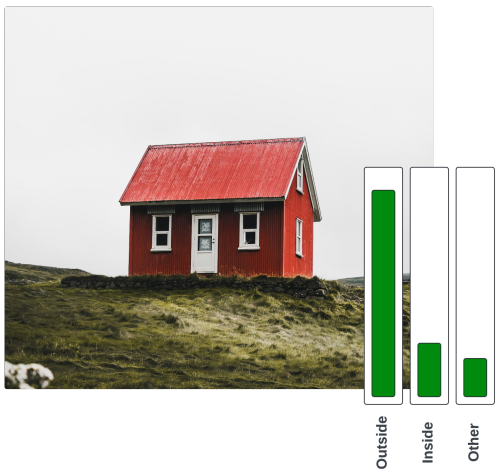


Figure 3.5: Stage 1

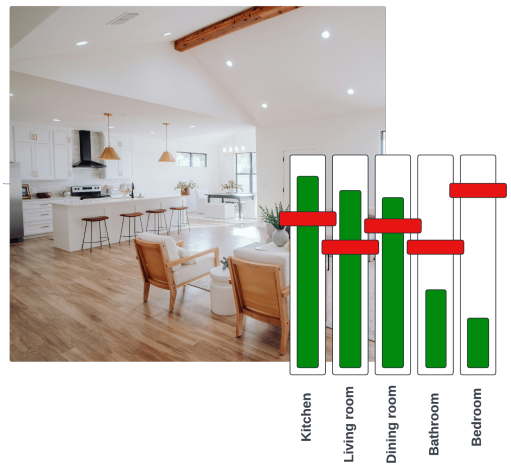


Figure 3.6: Stage 2

Figure 3.7: An instance of labeling process

Another technique used to improve the classification model is semi-supervised learning. This approach leverages pseudo-labels or high certainty predictions on unseen data from the initial model as additional training data in a second training stage. Consequently, it is deemed that a high-accuracy prediction above 90% in our study can be used as training data in a second training run.

Besides the five main room type labels in the second stage, there are three additional labels for aiding purposes, namely *Miscellaneous*, *Needs work*, and *Uncertain*, respectively. Needs work label is for images needing cropping or additional processing in case an image is a composite image of multiple images. Uncertain labels tag images that do not depict a room clearly, or the labeller is unsure. Miscellaneous labels are used to set up rooms not analysed in this study and are excluded; these rooms could, for example, be saunas or gyms. These additional tags makes it possible to indicate uncertainty and the required further work in parallel with labeling

the dataset. This is especially helpful in the form of non-furniture rooms that can serve multiple functions and are hard to label.

3.8 Data Pre-Processing

The metadata and the images are pre-processed, partly to create the required format for the models and partly to aid their convergence.

A frequently employed approach in similar studies [13] is to filter out sales for which the price deviates significantly from the other sales in the region in which it is sold or from an initial prediction. There can be multiple reasons for this price difference, such as data quality issues or failure to adhere to the market value conditions. Therefore, we determined that prices deviating from the initial estimate by more than a certain margin of error are unreliable and undesirable for the model to learn. As a result, this thesis excludes sales with prices that deviate from the original assessment by more than 80% in any direction.

Normalisation is another useful method that can make it easier for AI models to converge and learn representations, primarily because it is easier to grasp the ranges of the features [81]. Therefore, the numerical values used in the valuation model are normalised with a mean of zero and a standard division of one. This normalisation is based on the training data and then applied to the testing data in the test stage to continue out-of-sample learning.

Due to the high dimensional feature space in the valuation model, especially the categorical features connected to the location, feature reduction techniques are explored to limit the model from over-fitting to noisy variables. In this thesis, this is done by comparing the model trained on all provided variables with one that is only trained on the ones with higher feature importance using the SelectFromModel [85] function. The latter uses a pre-trained model on all the variables, keeping only the variables above the mean of the absolute feature importance for the specific model. This technique focuses on the more essential variables with the aim of a model that generalises better with only the more robust features. The model with the highest performance on the validation data is then chosen.

To streamline these pre-processing steps, two sci-kit pipelines [86] are created, with the first step handling the numerical and categorical features separately. The numerical features are standardised as described earlier with the provided StandardScaler [87]. At the same time, the categorical variables got one-hot encoded with the OneHotEncoder [88]. In the second pipeline, the additional feature selection is added, and only the features above the mean of importance are kept, as previously described. Figure 3.8 visually represents the pre-processing pipeline where X is the features and *y_pred* is the predicted market value.

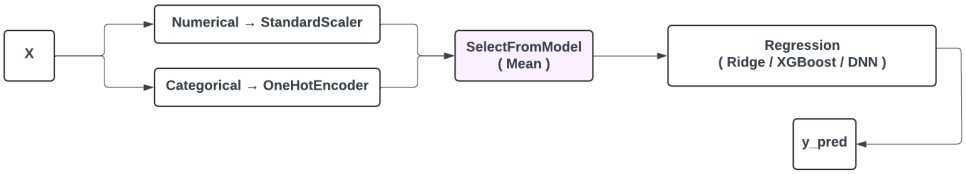


Figure 3.8: Pipeline for the AVM model

In the case of the normalisation for the DNNs used with the images, the pre-trained model usually has normalisation applied to the images during the training [89]–[91]. To work as expected, the same normalisation must be applied to the new data to give comparable and reasonable results. In PyTorch, these pre-processing steps are usually provided by the Transforms library [92] and applied during training. This is also the case for this thesis.

3.9 Deep Neural Networks

DNNs refer to neural networks with multiple hidden layers between the input and the output, making them deep. These deep models are primarily used during this thesis, and the following section outlines the methods used to train the DNN and highlights the best practices used.

These DNNs come in different forms to handle different kinds of data, whether it comes to data in succession, images, sounds, or inputs suited for standard feed-forward networks. However, the central concept is that these networks take these initial input signals and propagate them through a network, resulting in an output format designed to align with the task.

This thesis mainly works with two-dimensional and one-dimensional data. The two-dimensional inputs relate to the images, and the connected RGB colours relate to the red, green, and blue in the images. These are then used as input to find spatial patterns related to the task. A visual example of this can be seen in Figure 3.9

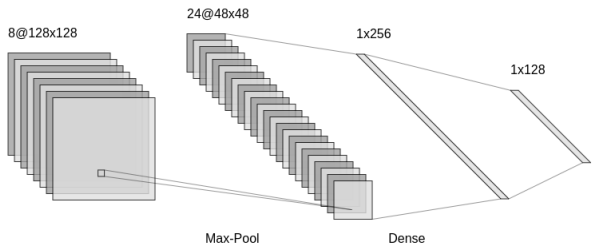


Figure 3.9: Neural Network with two-dimensional input

Meanwhile, the one-dimensional input relates to the numerical and categorical variables for the AVMs. Figure 3.10 shows a visual example of a model with a one-dimensional input.

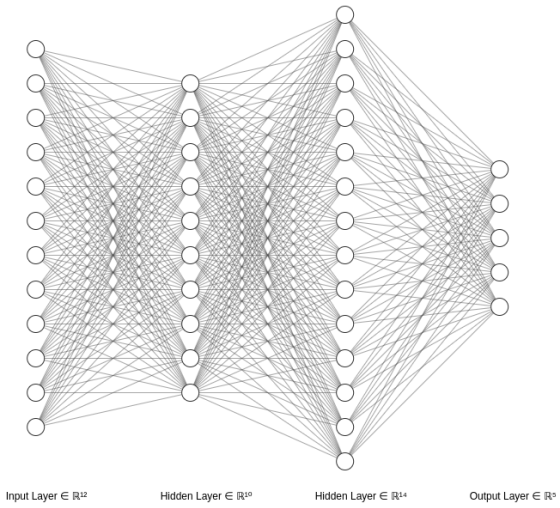


Figure 3.10: Neural Network with one-dimensional input

In the domain of AI models, and especially in the use of DNNs, the complexity of the models can be decided quite freely. This flexibility makes it a powerful tool that can be applied to various tasks. However, the trade-off between variance and bias on the task must be carefully assessed to decide on a suitable model, ensuring a suitable model complexity that has the ability to learn the patterns required in the task without overfitting to the noise within the data.

In addition to the model complexity selection, regularisation methods can reduce the model's tendency to over-fit on the training data. In this thesis, several regularisation methods are used. These include early stopping, which aims to stop the training phase when no new improvements are being made to the validation data [93]. Additionally, weight decay is used to penalise high weights during training to reduce weight changes on the noise. Furthermore, batch normalisation, which normalised the data between the layers, is incorporated into some of the models [94]. Dropout layers are also utilised to set a percentage of the neurons to zero during training to lower the reliance on certain neurons.

Another method used to leverage earlier models' learned patterns is transferring learning, where a model trained on a task has learned robust features to differentiate the data [95]. An example of this can be a room classifier on the Places365 dataset. This would then be able to be used in another task, replacing the end of the model with a new specific task. This new task is usually solved with a new sub-model on top referred to as the head of the model.

When training these pre-trained models, the best practice is to freeze or lock the base to retain the robust pattern learned in the previous task and only train the head [95]. This is done within a main training stage with a larger learning rate, followed by a fine-tuning training stage with a lower learning rate. Finally, even the base or backbone is added to the training with a minimal learning rate as a final task tuning of the whole model. For each stage, the model is trained until a set

maximum number of epochs is reached or until the early stopping halts the training.

Recent developments with training techniques and models within projects such as SimCLR [96] [97] and DINO [98] have shown the strength of self-supervised models. These models can learn robust features without the need for provided labeled data by augmenting an image and aiming to maximise the similarities between the original and the augmented image. The idea is that if the model has difficulty distinguishing the two, they are presumably similar.

In this thesis, these improvements are a perfect match due to the lack of provided labels and the focus on the differences in the images. The study used these improvements in the form of the provided pre-trained models [99] and in the form of training our own self-supervised base models for each room type with the DINO V1 [98] approach. All DINO V1 self-supervised trained models are trained using the default parameters provided, with the recommendation of 100 epochs for initial convergence. This limitation is set due to the runtime of training these models, requiring approximately two days per model, and uncertainly about whether the provided data size will be enough to generate an adequate model.

3.10 Visual Target Features

The following chapter focuses on visual feature extraction. This thesis explored three primary methods for feature extraction. The first is a binary classifier, where the decision between a desired and undesired attribute is assumed to be related to the percentage error or standard point. The second is an unsupervised approach, where clusters in connection with different models are assessed visually in relation to the architectural quality. The last approach uses the CLIP model, which can compare the similarity between images and a positive and negative description of a architectural quality, in a zero-shot fashion. Zero-shot learning refers to a scenario where the model can be applied to a task it was not trained on, and no examples were given with the task to the model [100].

3.10.1 Binary Classification

The binary classifier can be used as a simplified regression task where the magnitude is not directly related to the desired target variable but is assumed to be related. The provided magnitude is therefore ignored, and the target is converted into a simplified undesired or desired category related to the positive and negative sides. The model can then create its magnitude by analysing the common patterns on the positive and negative sides in the form of probability related to the desired or undesired classes.

This thesis uses two distributions to retrieve desirable and undesirable features. The first target, also used in earlier studies [3], is the percentage error of the prediction and the actual price without the visual features. Given the usage in the theory, it is assumed that the difference between the initial assessment and the market value depends on the lack of attributes related to the architectural quality that can be

observed in the image. The more images in the undesired class that share similar patterns, the stronger the predictor of an undesired feature.

The second target is the standard point, which adds a Swedish-specific approach. This score only exists for the houses. It is a condition measurement used as part of the housing declaration. This score assesses multiple factors, such as the aspect of the property's exterior, energy management, kitchen condition, sanitation, and other interior features that align with the condition and function [84].

However, while these scores are not only based on interior features that can be seen, within this thesis, it is assumed that a home with a high standard point score or a high percentage error has visual features in the interior of the property that indicate higher architectural quality.

Figures 3.11 and 3.12 show the percentage error and the standard points distribution used during training, which seem to follow a normal distribution. Zero is the divider in the percentage error case, while the empirical mean is the divider between lower and higher in the standard point scenario. To focus on the more distinct differences, the sales outside of the absolute ten percentages are used in the percentage sale case.

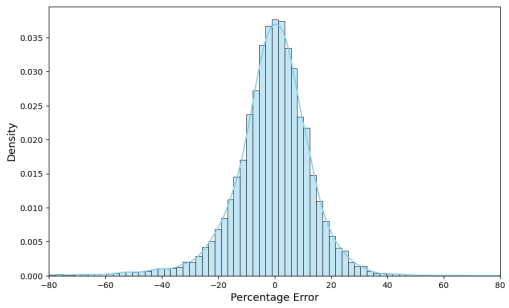


Figure 3.11: Histogram of the percentage error

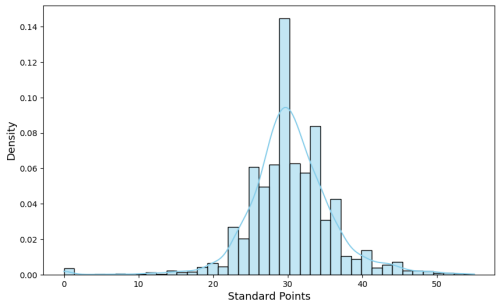


Figure 3.12: Histogram of the standard point

To evaluate and score the binary classifier, an *area under the curve (AUC)* score is generated for each model to compare their ability to quantify the desired and undesired features. The AUC score is obtained from the *receiver operating characteristic (ROC) curve*, which shows the *true positive rate (TPR)* compared against the *false positive rate (FPR)*. The equations for FPR and TPR can be seen below.

$$\text{TPR} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{3.1}$$

$$\text{FPR} = \frac{\text{False Positives}}{\text{False Positives} + \text{True Negatives}} \tag{3.2}$$

Consequentially, the common method to calculate the AUC is using the trapezoidal rule. This rule approximates the area under the ROC curve by dividing that area into multiple trapezoids, with vertical lines for FPR values and horizontal lines for TPR values. After that, the area is calculated by summing the areas of these

trapezoids [101]. During this project, sci-kit learn *roc_auc_score* [102] function implementation is used to get the AUC score.

Due to a late improvement, these improvements are not assessed in the form of the AVM models but rather by themselves compared to the validation data. AUC is a way to score how well the model distinguishes the classes. Generally, a score of 1 is considered perfect class separation, and a score of 0.8 is regarded as a good separation. However, a score of 0.5 shows no ability to separate the groups [101].

3.10.2 Clustering

The last method used for visual feature extraction is clustering. In this method, a pre-trained model generates a high-dimensional vector that is then used to find clusters of images based on similar visual traits. The aim here is to find visually interpretable clusters that relate to architectural qualities.

Multiple models are used to generate these high-dimensional vectors. Firstly, the ViT Base model with 14 patches provided by the DINO v2 project [103] [104], which has been trained on ImageNet [51] using a self-supervised approach, is used. Secondly, the self-supervised model that is trained within this thesis on each room type is tried. Finally, two pre-trained CNN models, VGG and ResNet50, that are pre-trained on the Places365 dataset are utilised.

These high-dimensional vectors are then normalised and clustered with the *K-Means* algorithm based on the *Euclidean distance* seen below:

$$\sqrt{\sum_{i=1}^{n}(A_i - B_i)^2} \tag{3.3}$$

In this equation, A and B are the vectors in n-dimensional space, and A_i and B_i indicate components of the A and B vectors, respectively.

This process is repeated with a range of different numbers of clusters (k-values) between 2 and 10. This range is chosen to look for more significant clusters and assess more models within the thesis.

Getting a good indication for well-divided clusters might vary depending on the project framework, domain, and data. For this project's framework, the *Davies-Bouldin* score is used as the primary indicator to determine the best number of clusters to focus on. This effectively differentiates between the distinct clusters and ensures they are well-separated and compact [105]. This is used as an indicator for what clusters to explore more.

To select the cluster feature to be extracted and included in the AVM, a visual inspection is performed with the aim to understand the clusters visually and relate them to our interpretation of the architectural qualities. This involved randomly sampling five images from each cluster, a process repeated three times to ensure a diverse representation of the perceived quality found in the images. This rigorous approach lowers the chances of the characteristics being found simply due to chance.

3.10.3 Contrastive Language-Image Pre-Training

CLIP is a model developed by the OpenAI team that has been trained to tie together the ViT encoding of images to the text encoded image descriptions with the help of cosine similarities, which have earlier been used within the field of Natural Language Processing (NLP) to match document types [10]. This returns a score between minus one and one, representing the similarity between vectors A and B, as shown below. In addition, the representation of $||X||$ in the equation refers to the Euclidean norm for vectors.

$$\text{Cosine Similarity}(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} \tag{3.4}$$

In this study, the clip model is used to score the architectural qualities by providing a textual description and then matching it with the images in a zero-shot fashion. A positive and a negative version of the targeted feature is used to provide a range of results. Figure 3.13 shows an example of extracting a score for the room's utility to move around by matching the spaciousness in text format with the room type image. This involves inputting two sentences in the text encoder and taking the positive score minus the negative score as the saved score for the feature.

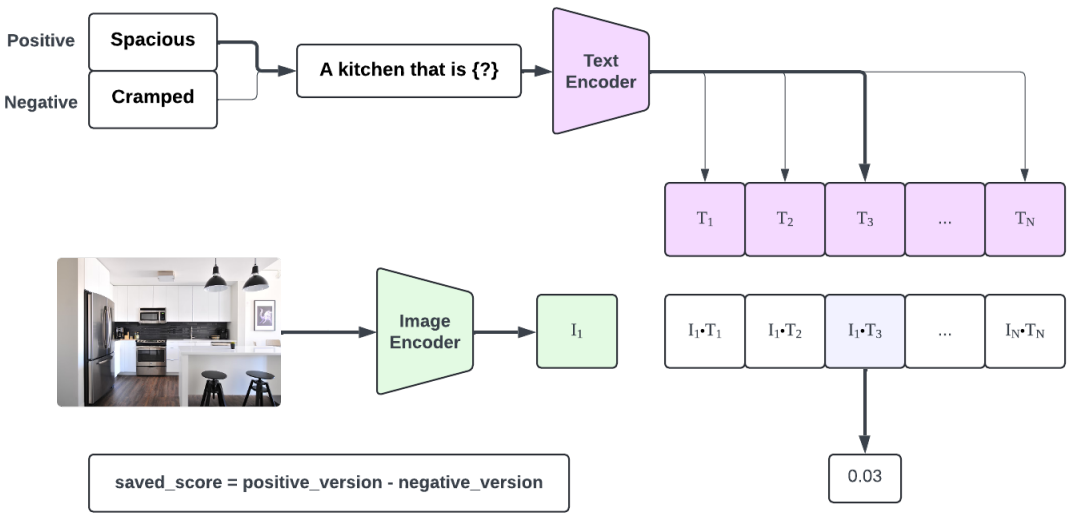


Figure 3.13: Use of the CLIP model to generate spaciousness score [106]

These text versions of the positive and negative architectural qualities are primarily based on examples found in the literature, with some additions from our understanding of what is considered desirable and undesirable within the room types.

3.11 Utilising Visual Features in the Model

After the corresponding model extracted the visual features, these features are labelled with a name that tied them to the model and targeted features. They are then

added to the enriched version of the base features with the model-specific features. The scores are averaged, when multiple images associated with the same room type and numerical visual features are present. This is a typical approach when using multiple images connected to the same feature [13], [16]. However, in the case of a categorical variable, the presence of one is enough for it to be valid for the entire sale. For example, one bathroom image with a bathtub is enough for the "bathroom has bathtub" feature.

In the case of missing images of a room type, which results in missing features, the average of the features is used to fill in the missing values. This method is also used in a similar study, and it is a normal way of handling missing values without completely excluding the rows [4]. This is only done when a row has any visual features. However, if there are only non-visual features, the sale is excluded.

3.12 Automated Valuation Model

A crucial part of our study is the AVM, which highlights the importance of our visual features related to the price. Within this thesis, three base models are chosen due to their previous use as baseline models and the ease of extracting the importance of the feature input.

An advantage of linear regression models and XGBoost [23] is that they are typically used as baselines. This is primarily because they can also provide a feature importance that shows the weight or importance of a feature [34].

Firstly, Ridge regression is a model that uses a regularisation technique to improve the model's accuracy. During this regularisation, a penalty term proportional to the square of these coefficients is added to the loss function. This makes the model better at generalising the data, thus controlling the model's complexity and reducing the model's tendency to overfit the data. This is chosen over normal Linear Regression in our case because it minimises the chance of overfitting problem on data with high variance. However, they share the same fundamental basis, which is the linear assumption, and can also be used to show the weight of the feature.

Secondly, the XGBoost is an implementation of gradient-boosted decision trees. It aims to improve prediction accuracy by using an ensemble of trees [107]. To do this, it corrects errors from previous trees iteratively. A key difference to the linear methods is that it can catch more complex patterns in the datasets.

Lastly, neural networks, or DNNs, which can have one or multiple neuron layers, can process data through these neuron layers to recognize patterns in the dataset. It can learn non-linear relationships in the dataset, making it capable of finding more complex relations and features available. In the context of AVM, it provides the benefit of being able to processing complex inputs such as images as part of the same model to obtain several patterns that might affect the valuation of the properties.

Both of the neural network AVMs used are trained in five stages with a *mean square error loss (MSELoss)* [108] loss function seen below.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(t_i - p_i)^2 \tag{3.5}$$

where N represents the batch size, t_i is the actual or true value and p_i is the predicted value. It was trained with an initial learning rate of 0.01 that is then divided by ten after each run until the final run of 0.00001 with a base size of 512, and an early stopping with patience of 4 and weight decay regulation of 0.00046. The dropout layers in the model are halved between each iteration, starting with 10%.

In the selection of hyper-parameters for the AVMs, a grid search is performed to find the highest-scoring combination of parameters. It is achieved by running cross-validation on the training data with different pre-decided ranges for the different parameters. Scoring the parameters on the lowest MAPE score achieved.

The hyper-parameters that are explored in this thesis related to the Ridge model is the *alpha (α)* value, which refers to the constant that controls the regularisation strength by being multiplied with the L2 term.

For the neural network, the training regularisation hyperparameter of weight decay, the learning rate during the training stages, and patience for early stopping are explored.

For the XGBoost model, the following hyperparameters are explored to prevent overfitting. These include `colsample_bytree` that indicates the fraction of features per tree, `learning_rate` controls the training step size, `max_depth` sets maximum tree depth, `min_child_weight` ensures minimum instance weight in child, `n_estimators` sets the number of trees, `subsample` uses a fraction of data for each tree to generalise better, `gamma` sets the minimum loss reduction required for a split which makes the model conservative, and `alpha` applies a regularisation to prevent overfitting for the XGBoost model. The resulting hyper-parameters used are described with the model in the result.

3.13 Scores

The final analysis of this thesis focuses on the AVM error rates with and without the newly integrated features to capture the importance of the visual features regarding market value prediction.

One important rule when validating models is to use an out-of-sample prediction approach, where all models are scored on unseen data during the training stage with the aim of capturing how well they would perform in a real-case scenario. Due to the various models used and the diverse models that contribute to the final visual feature pool, it is decided to split the dataset in the pre-processing stage. This separation prevents these different splits from causing in-sample bias when extracting visual features.

One of these metrics is *Mean Absolute Error (MAE)* [109], which highlights the average difference in the error. In our case, it shows the absolute amount of *Swedish*

Krona (SEK) that the predictions differ from the actual selling price. The formulation of MAE can be seen below, where the p_i is the i:th prediction and t_i is the i:th actual value.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|t_i - p_i| \tag{3.6}$$

Another more easily comprehensive error metric is the *Mean Absolute Percentage Error (MAPE)* score [110], which shows how much an estimation is wrong on average in the percentage of the actual value. This makes it easier to get comparable results between regions with different prices. For example, a 200,000 SEK error on a 200,000 SEK property differs from a 200,000 SEK error on a 2,000,000 SEK property. The formulation of MAPE can be seen below.

$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{t_i - p_i}{t_i}\right| \tag{3.7}$$

Furthermore, another metric is chosen to highlight the worst prediction in proportion. This metric generates a score for the MAPE on the 10% worst predictions. This highlights how far off the model is on the worse predictions. Additionally, R^2 is the coefficient of determination of a regression model. Its value shows the proportion of variance in the dependent variable, which is the selling price in our case [111]. Finally, the *Median Error Rate* [112] is used to highlight the centre of the errors. This method is robust against outliers because it excludes them from the actual score, unlike the averaging scores method. Instead, as the name suggests, it shows the median error.

These results in five scores that highlight different metrics to give a broader picture of the differences between before-and-after visual features and where improvements are made. However, the main focus of this thesis is the MAPE score due to its ease of interpretation concerning the different property types.

Lastly, the feature's importance is assessed. In the case of Ridge Regression, the feature's importance is ordered by the absolute coefficient to focus on magnitude and not solely on positive features. Additionally, two usual alternatives for XGBoost feature importance are gain and weight. Gain indicates the contribution makes by the feature, and weight indicates the frequency in which it is used [113]. This thesis focuses on improvements in the form of gain rather than usage to align with the goal of reducing the uncertainty.

4

Results

The following chapter shows the study's results. First, the outcomes of the pre-processing and the performance of the room classifier will be highlighted. Next, the results of the self-supervised attention will be compared visually. Then, the results of the visual feature extraction process will be displayed. Finally, the AVM score and the importance of the features of the models will be exhibited.

4.1 Classification of Images

The hierarchical classification approach to label the rooms began with separating the interior images. Different models were compared, leading to the selection of a ViTS14 with the pre-trained weights from DINO V2 [103], [104]. The model was trained using a *cross entropy (CE)* loss [114] function where its equations in the form of binary and multi-class can be seen below.

$$\text{CE (Binary Classification)} = -\left(y \log(p) + (1-y) \log(1-p)\right) \tag{4.1}$$

In the binary version, y denotes the actual label which is either 0 for false or 1 for true and p represents the predicted probability that the label is true.

$$\text{CE (Multi-class Classification)} = -\sum_{i=1}^{N} y_i \log(p_i) \tag{4.2}$$

In the multi-class version, N represents the number of classes, y_i indicates the binary indicator where 1 indicates the correct classification for class label i and 0 otherwise. Also, p_i denotes the predicted probability for class i.

It was trained with an initial learning rate of 0.001 and fine-tuning at 0.0001. The head of this model can be found in Appendix B, as shown in Figure B.4.

After pseudo-labelling and re-training using the same loss function and step sizes, Figure 4.1 shows the final confusion matrix on the test dataset. It highlights an excellent ability to distinguish the classes, with a few instances where the model was confused.

For the second labelling stage, the same ViT base model was used with a similar head, but ending with a Sigmoid function and one neuron per room type, as shown

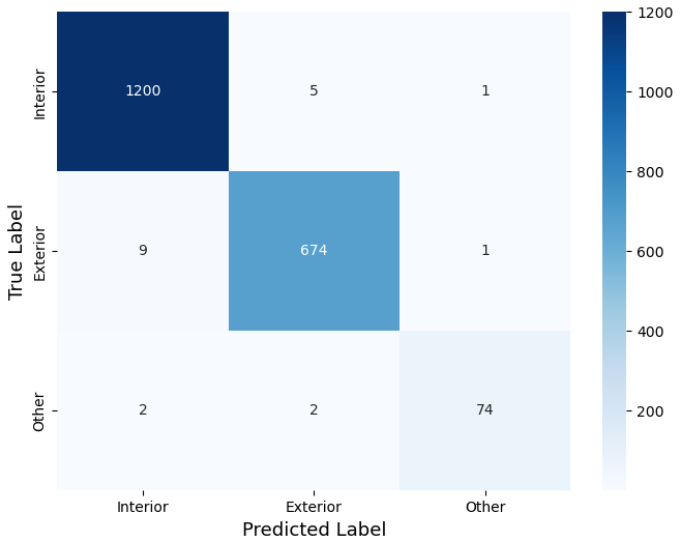


Figure 4.1: Confusion matrix of interior and exterior classifications

in Figure B.5 in Appendix B. It was trained with an initial learning rate of 0.001, which changed to 0.00001 during fine-tuning, and an early stop patience of 4 while the maximum number of epochs was set to 100.

Thereafter, the model generated the thresholds indicating that an image corresponds to a particular room type. Table 4.1 shows the threshold selected by finding the best threshold according to the highest F1 score on the validation data for each room type. Consequentially, these thresholds indicates a high F1 score and ability to differentiate the room types.

Table 4.1: Best thresholds and F1 scores for different rooms

Room	**Best Threshold**	**Best F1 Score**
Bathroom	0.46	0.988
Bedroom	0.43	0.947
Dining room	0.56	0.906
Kitchen	0.46	0.955
Living room	0.44	0.904

The trained model was run on the entire unlabeled dataset, allowing the images with only one predicted room to be extracted and used for the study. The resulting number of images within each room type can be seen in Figure 4.2, showing a significantly lower number of images for the dining room compared to the other classes.

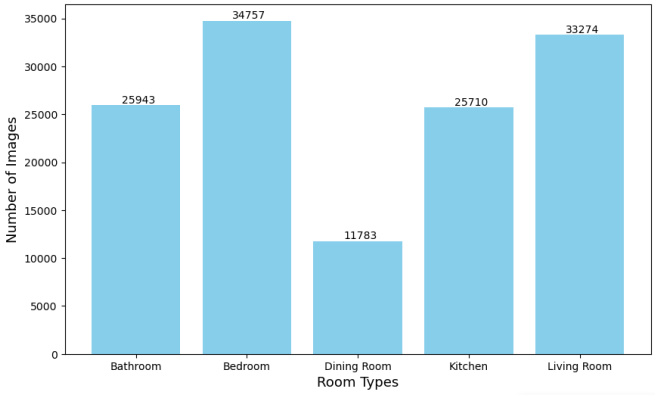


Figure 4.2: Number of images in each room type used in the study

4.2 Self-Supervised Models

In this study, one self-supervised DINO V1 model was created for each room type in order to obtain a model that was more tailored to the room type. A comparison of the difference in the attention from the provided pre-trained weights of the small ViT model with a patch size of 8 can be observed in Figure 4.3. In contrast, Figure 4.4 shows the attention from the same base model that had been trained on our kitchen images applied to an out-of-sample kitchen image. It can be seen that the pre-trained model exhibits clear attention to the kitchen parts, while the self-trained model seems to focus more broadly on the room.

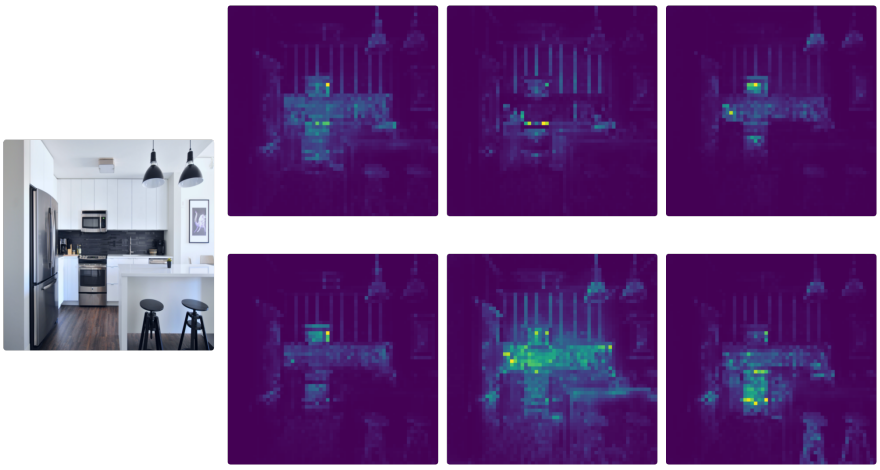


Figure 4.3: Pre-trained ViT model attention on a kitchen example

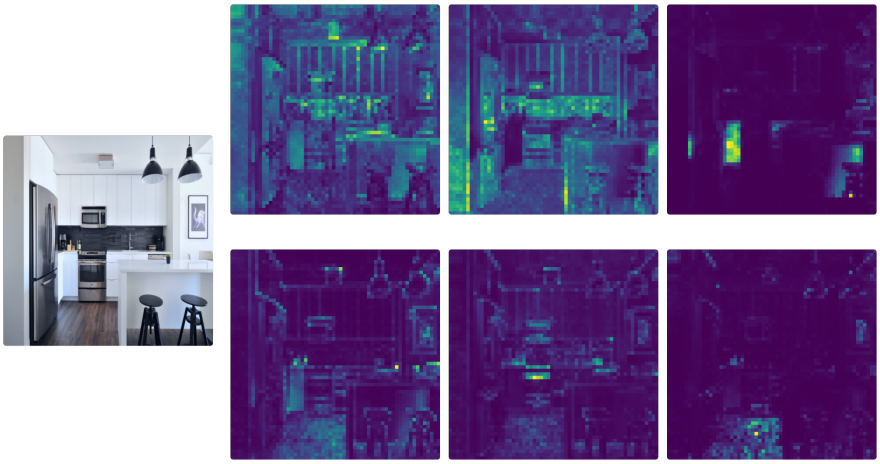


Figure 4.4: Our Self-supervised ViT model attention on a kitchen example

4.3 Feature Extraction

This section shows the performance of the binary classifiers and AVMs, the chosen visual features, and the targeted architectural quality.

4.3.1 Binary Classifier

Starting with the binary classifier result in the form of an AUC score for each room type and the models used, the final models compared were the VGG16 and ResNet50, with weights from being pre-trained on Places365. The DINO model utilised was the self-supervised version of the room type trained within the thesis, while the pre-trained version is the DINO V2 ViTb14 model. The head of the used model can be found in Figure B.3 in Appendix B. Meanwhile, Table 4.2 shows that the pre-trained model from DINO v2 outperforms the other models while still being below the desired score of 0.8 for a clear separation for all the room types.

Table 4.2: Area under the curve score for the room types on percentage error

Model Name	Kitchen	Living room	Bathroom	Bedroom	Dining room
VGG16	0.68	0.65	0.62	0.62	0.62
ResNet50	0.68	0.66	0.63	0.65	0.61
ViT (Pre-trained)	**0.74**	**0.68**	**0.70**	**0.68**	**0.67**
ViT (Self-supervised)	0.61	0.65	0.64	0.60	0.58

Secondly, Table 4.3 shows the model AUC performance of the same models with the same head, below or above the average standard points. This indicates poor performance in utilising the standard points compared to the percentage error.

Table 4.3: Area under the curve score for each room type on standard point

Model Name	Kitchen	Living room	Bathroom	Bedroom	Dining room
VGG16	0.53	0.53	0.54	0.53	0.51
ResNet50	0.57	0.54	0.55	0.56	0.48
ViT (Pre-trained)	0.56	0.58	0.57	0.57	0.54
ViT (Self-supervised)	0.54	0.56	0.51	0.55	0.54

4.3.2 Clustering Features Found

As a result of the clustering process, the features deemed clear and connected to the architectural qualities were selected through a manual visual inspection. Multiple models were used, including our self-supervised and pre-trained DINO models, VGG16 and ResNet50. However, only the DINO V2 model created clusters deemed easily distinguishable that could be connected to the architectural qualities. The clusters and short names for each of the features connected to the room type can be observed in Table 4.4.

Table 4.4: Visual clustering features selected with beauty (B) and utility (U)

Rooms	Features From Clusters		
Bathroom	Bathtub [U]	Glass Shower [B]	Washing Machines [U]
Bedroom	Big Windows (Bright room) [B]	Empty Room (Unfurnished) [B]	Clear Placement of a Bed [U, B]
Living Room	Clear Place For Television [U, B]	Fireplace [U, B]	Empty Room (Unfurnished) [B]
Kitchen	Full Oven [U, B]	-	-
Dining Room	-	-	-

In this table, the conceived architectural qualities connected to the found clusters are highlighted. The label *U*, refers to the *Utility*, the functional use of the property, and *B* refers to the *Beauty*, which is about the aesthetic appearance and luxury of the property. These are our interpretations tied to the architectural qualities.

4.3.3 CLIP Features Explored

For the CLIP features, architectural qualities were extracted using two textual versions, in the form of a positive and negative variant. Table 4.5 displays a few examples from the features used to create the text encoding on which the images within the CLIP model were scored, with the targeted architectural quality connected. For a complete list of features and the positive and negative statements used, see Appendix A.

Table 4.5: Examples of the CLIP features selected

Architectural Quality	Code	Room Type	Positive (Pos)	Negative (Neg)
Stability	KI_01	Kitchen	Kitchen shows the walls and ceiling in mint condition	Kitchen with visible damages and cracks on walls and ceiling
Utility	KI_05	Kitchen	Kitchen includes multiple useful built-ins	Kitchen without built-ins
...	...	...	...	....
Stability	LI_07	Living Room	Living room free from water damage or mold	Living Room with water damage or mold
Beauty	LI_12	Living Room	Aesthetically pleasing living room	Displeasing living room
...	...	...	...	....
Stability	BA_07	Bathroom	Bathroom free from water damage or mold	Bathroom with water damage or mold
Beauty	BA_15	Bathroom	Bathroom with a lot of natural light	Bathroom with only artificial light
...	...	...	...	....
Stability	DI_03	Dining Room	Dining room features high-quality flooring	Dining Room with damaged flooring
Utility	DI_11	Dining Room	Dining room with a seamlessly flows into adjacent spaces	Cramped dining room with bad layout
...	...	...	...	....
Utility	BE_02	Bedroom	Bedroom shows the walls and ceiling in mint condition	Bedroom with visible damages and cracks on walls and ceiling
Beauty	BE_10	Bedroom	Bedroom showcases unique architectural details	Bedroom showcases regular architectural details

4.4 Sales

After the previous classification of room types and visual feature extraction, the sales where at least one visual feature was added were kept to be used in the AVM part. Figure 4.5 shows the initial data in blue and the remaining data used in the study in red. It can be observed that the number of house sales in this study is considerably less than the number of apartments. Additionally, the final dataset was reduced to 7580 for apartments and 3252 for houses.

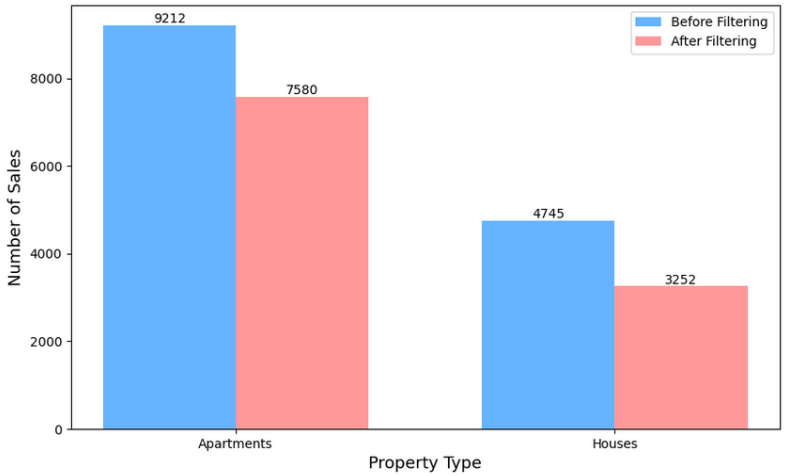


Figure 4.5: Sales of property type before and after filtering

The resulting sale distribution within the Uppsala county region can be seen in Figure 4.6 for the apartments and Figure 4.7 for the houses. This indicates a wider

spread of houses over the region, where apartments are primarily clustered closer to urban areas.

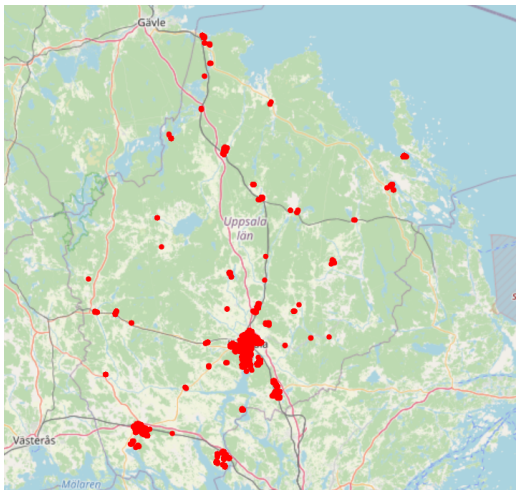


Figure 4.6: Apartment sales in Uppsala County

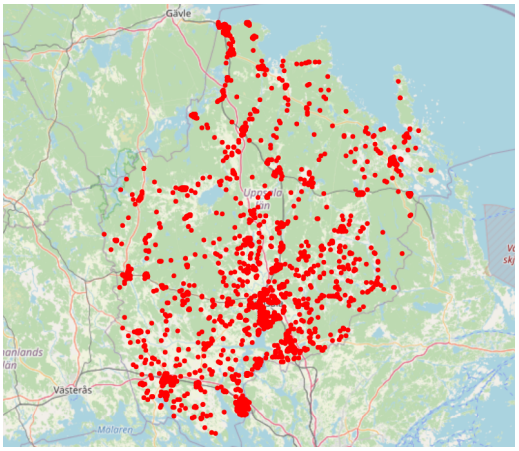


Figure 4.7: House sales in Uppsala County

4.5 Automated Valuation Model

The following section highlights the AVMs' accuracy with and without the visual features for apartments and houses.

The main comparison of the improvements is conducted on the MAPE score. A paired t-test is conducted for each model with a 10-fold cross-validation between the model with and without the visual features.

The hypotheses can be observed below:

- H_0 (Null hypothesis): No difference between the MAPEs with and without the added features.
- H_1 (Alternative hypothesis): Adding features decreases the value of MAPE.

Since adding visual features is expected to improve the valuation in form of MAPE score, a significant value of 0.05 is chosen. This is a commonly used value for statistical testing [115].

In our study, we focused on the *individual* significance level instead of the overall significance level that we would get by applying corrections for multiple testing corrections, such as Bonferroni corrections [116]. This choice results in a higher chance of rejecting some of the several null hypotheses for the models only by chance, even though they were true.

4.5.1 Apartments

The analysis and evaluation of the apartment models begin with the Ridge models, where an α value of 1 was used. This value was selected based on a cross-

validation test showing the lowest *mean squared error* on the training data. This hyper-parameter was used for all the models while being selected on the model without visual features.

Table 4.6 shows improvements in the test data with the visual features of MAPE, where the mean for cluster features is lowered by -0.1, and the CLIP features reduced the mean MAPE by -0.48. Meanwhile, the mean difference in cross-validation for CLIP is slightly higher (-0.59). Also, the connected p-values in Table 4.7 show that both improvements reject the null hypothesis.

Table 4.6: Performance metrics on Ridge for apartment AVM

Model Name	**MAPE (%)**	$\boldsymbol{R^2}$	**MAE**	**MAPE (10% Worst)**	**Median AE**
Base Model	8.20	0.90	175855.38	25.90	126163.77
Clusters	8.10	0.91	173922.28	25.20	125833.46
CLIP	7.72	0.91	167711.34	24.02	117608.73

Table 4.7: Paired sample t-Test results for apartment AVM with Ridge

Description	**Mean Difference**	**Standard Error**	**t-value**	**df**	**p-value**
Baseline vs Cluster	-0.10	0.03	-3.74	9	0.0046
Baseline vs CLIP	-0.59	0.06	-10.34	9	0.0000

Secondly, the XGBoost model for the apartment used the highest scoring parameters of `colsample_bytree=0.8`, `learning_rate=0.1`, `max_depth=5`, `min_child_weight=2`, `n_estimators=1000`, `subsample=0.7`, `gamma=0.2`, and `alpha=0.1`.

Table 4.8 indicates that the CLIP model shows the best overall performance with the lowest MAPE, MAE and Median AE scores along with an on pair R^2 score with the base model. Meanwhile, the cluster model performed slightly worse than the base model on all the metrics except the mean average error. In the paired t-test seen in Table 4.9, it can be observed that the cluster achieved a minor improvement on the mean MAPE while not rejecting the null hypothesis. However, using CLIP features reduces it by 0.3% and rejects the null hypothesis with a p-value of 0.0023.

Table 4.8: Performance metrics on XGBoost for apartment AVM

Model Name	**MAPE (%)**	$\boldsymbol{R^2}$	**MAE**	**MAPE (10% Worst)**	**Median AE**
Base Model	8.66	0.90	185831.29	26.17	134109.88
Clusters	8.74	0.89	186773.93	26.86	132995.68
CLIP	8.27	0.90	179081.34	25.75	126286.31

Table 4.9: Paired sample t-Test results for XGBoost

Description	**Mean Difference**	**Standard Error**	**t-value**	**df**	**p-value**
Baseline vs Cluster	-0.06	0.03	-2.13	9	0.0624
Baseline vs CLIP	-0.30	0.07	-4.22	9	0.0023

Thirdly, in the case of the neural networks for the apartments, the final model structure can be seen in Appendix B, as shown in Figure B.1.

The final scores on the AVM task can be observed in Table 4.10, and the connected paired t-test scores for significance are shown in Table 4.11.

Table 4.10: Performance metrics on Neural Network for apartment AVM

Model Name	MAPE (%)	R^2	MAE	MAPE (10% Worst)	Median AE
Base Model	7.74	0.92	167542.12	23.79	118265.12
Clusters	7.78	0.91	168656.97	24.09	117582.35
CLIP	7.77	0.91	171808.74	23.69	117240.44

Table 4.11: Paired sample t-Test results on Neural Network for apartment

Description	Mean Difference	Standard Error	t-value	df	p-value
Baseline vs Cluster	0.01	0.08	0.12	9	0.9051
Baseline vs CLIP	-0.25	0.12	-1.98	9	0.0786

4.5.2 Houses

The analysis and evaluation of the housing models begin with the Ridge models, where an α value of 10 was used due to resulting in the lowest MAPE score in cross validation on the training data. Table 4.12 shows improvements with the visual features of MAPE, and Table 4.13 shows the results of the connected paired t-test between the base model and the ones with visual features.

Both visual additions lowered the MAPE and had a p-value below 0.05. However, in the cross-validation, this reduction is only 0.1% for the clusters and 0.6% for the CLIP.

Table 4.12: Performance metrics on Ridge for house AVM

Model Name	MAPE (%)	R^2	MAE	MAPE (10% Worst)	Median AE
Base Model	17.41	0.788	562406	58.95	401411
Clusters	17.05	0.798	547299	57.16	374786
CLIP	15.02	0.827	502663	50.28	348987

Table 4.13: Paired sample t-Test Results for house Ridge

Description	Mean Difference	Standard Error	t-value	df	p-value
Baseline vs Cluster	-0.10	0.03	-3.47	9	0.0070
Baseline vs CLIP	-0.60	0.05	-12.68	9	0.0000

Secondly, the XGBoost model for the house used the parameters of `colsample_bytree=0.8`, `learning_rate=0.1`, `max_depth=5`,

`min_child_weight=2`, `n_estimators=1000`, `subsample=0.7`, `gamma=0.2`, and `alpha=0.1`.

The comparison of the models with and without the visual features can be observed in Table 4.14. It shows a worse score with the use of the cluster features and a reduction of 1.58% with the CLIP on the testing data.

Furthermore, the paired t-test result in Table 4.15 shows that the cluster had a slight reduction in the mean of MAPE. However, It does not reject the null hypothesis. In the case of the addition CLIP feature, a MAPE reduction of 0.91 is observed with a p-value of 0.0312 that still rejects the null hypothesis.

Table 4.14: Performance metrics on XGBoost for house AVM

Model Name	**MAPE (%)**	R^2	**MAE**	**MAPE (10% Worst)**	**Median AE**
Base Model	16.45	0.808	530301.61	55.245	389171.50
Clusters	16.60	0.795	543524.981	55.75	378538.50
CLIP	14.87	0.829	494919.092	49.34	351693.50

Table 4.15: Paired sample t-Test results for house XGBoost

Description	**Mean Difference**	**Standard Error**	**t-value**	**df**	**p-value**
Baseline vs Cluster	-0.26	0.14	-1.90	9	0.0906
Baseline vs CLIP	-0.91	0.36	-2.55	9	0.0312

Thirdly, in the case of the neural networks for houses, the same model utilised in the apartment model was used, and it can be observed in Appendix B, as shown in Figure B.2.

The final scores of the house DNN on the AVM task can be observed in Table 4.16, and the connected paired t-test scores for significance are shown in Table 4.17. These results demonstrate that adding the cluster features were not able to reject the null hypothesis. However, the CLIP version showed a statistically significant reduction of 4% while improving all the provided metrics on the test data.

Table 4.16: Performance metrics on Neural Network for house AVM

Model Name	**MAPE (%)**	R^2	**MAE**	**MAPE (10% Worst)**	**Median AE**
Base Model	17.87	0.756	601334.22	56.535	426081.40
Clusters	17.13	0.772	576544.44	54.617	405768.35
CLIP	14.88	0.804	520958.18	45.934	358481.52

Table 4.17: Paired sample t-Test results for house Neural Network MAPE

Description	**Mean Difference**	**Standard Error**	**t-value**	**df**	**p-value**
Baseline vs Cluster	-0.18	0.29	-0.62	9	0.5487
Baseline vs CLIP	-4.03	0.39	-10.43	9	0.0000

4.5.3 Feature Importance

This section will investigate the importance of features for the models. The baseline model without visual features will be explored first, followed by the visual features.

4.5.3.1 Apartment

Starting with the features that are important for the apartments, Figure 4.8 shows the feature importance in the Ridge model with base features. It highlights the importance of the living area, locality, housing cooperatives and rooms.

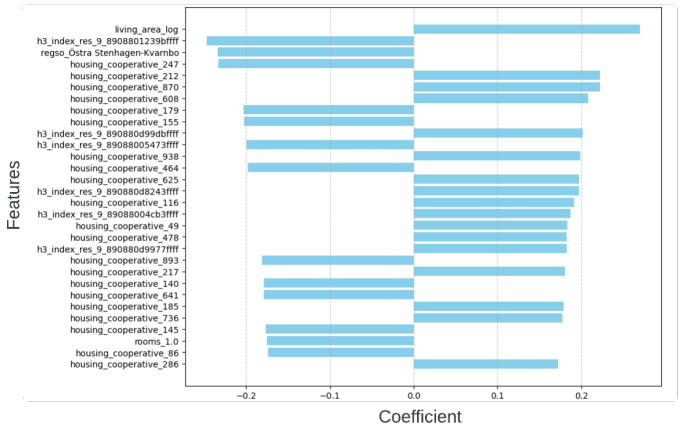


Figure 4.8: Top 30 features - apartment AVM [Ridge] - With base features

Figure 4.9 shows the highest scoring features on the XGBoost model on the apartments with base features. The most contributing features are those connected to the locality, followed by property-specific features such as the number of rooms and living spaces.

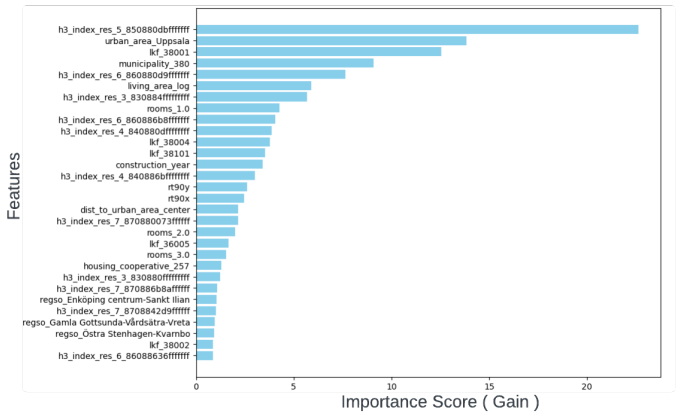


Figure 4.9: Top 30 features - apartment AVM [XGBoost] - With base features

4.5.3.2 House

This section will show the most important features of the model for the houses. Figure 4.10 illustrates the importance of features in the house AVM model of Ridge with base features. Once again, the locality features are frequently present and high-scoring. Subsequently, the property's living area is followed by the one-room apartments and the construction year for the property-related features.

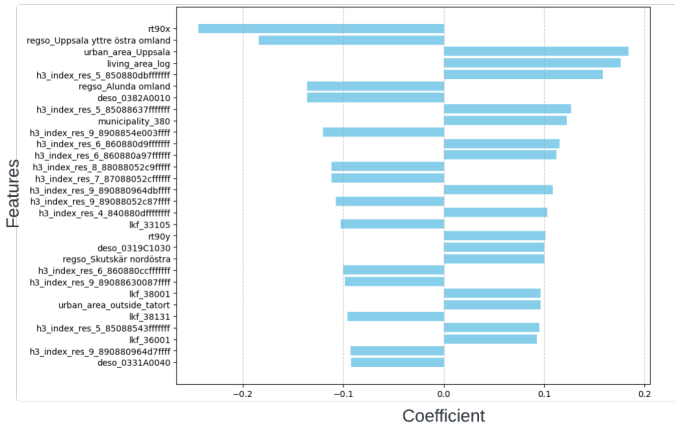


Figure 4.10: Top 30 features - house AVM [Ridge] - with base features

For the XGBoost model on the houses with base features, Figure 4.11 shows the highest-scoring features, with regional features once again at the top in the form of the coordinates, urban area, H3 index and municipality. Furthermore, the living area and the construction year are also part of the top 30 features.

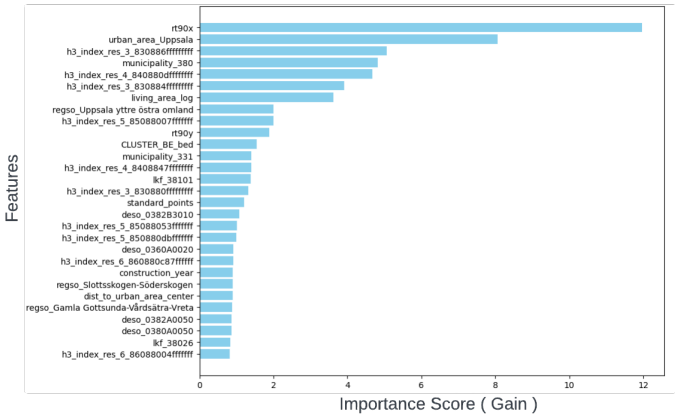


Figure 4.11: Top 30 features - house AVM [XGBoost] - with base features

4.5.4 Visual Features Importance

The focus will now shift to the added visual features' importance in the models. However, models that did not manage to reject the null hypothesis will be excluded, as they were unable to disprove that the observed changes happened by chance.

The achieved mean improvement for MAPE in the paired t-test of the base model will also be noted in relation to the graphs to highlight the actual improvement the models demonstrated in our experiments.

4.5.4.1 Apartment

Starting with the added cluster features on the Ridge model for apartments, which achieved a mean difference of -0.1, Figure 4.12 highlights only the cluster features for the previous model. As can be seen, all the binary features have relatively low coefficients compared to the top features. However, the empty bedroom (BE_no_furniture) seem to be the highest weighted cluster, followed by the glass shower in the bathroom (BA_glass_shower) and the fireplace in the living room (LI_fireplace).

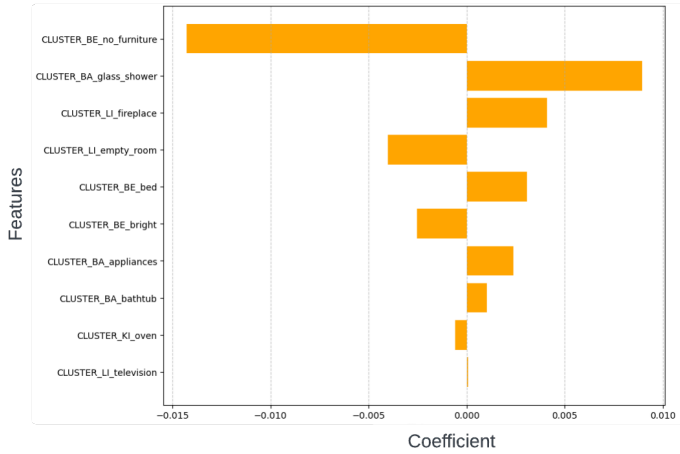


Figure 4.12: Top features - apartment AVM [Ridge] - with only cluster features

In the case of the apartment Ridge model with an improvement of -0.59, Figure 4.13 shows the feature importance of the Ridge model with only CLIP features for the Apartment AVM. It can be seen that most features, such as BA_03 (floor quality) and KI_13 (related to renovation), contribute positively to the Ridge model. However, a few features, such as BE_04 (related to aesthetics) contribute negatively to the model on the valuation of apartments.

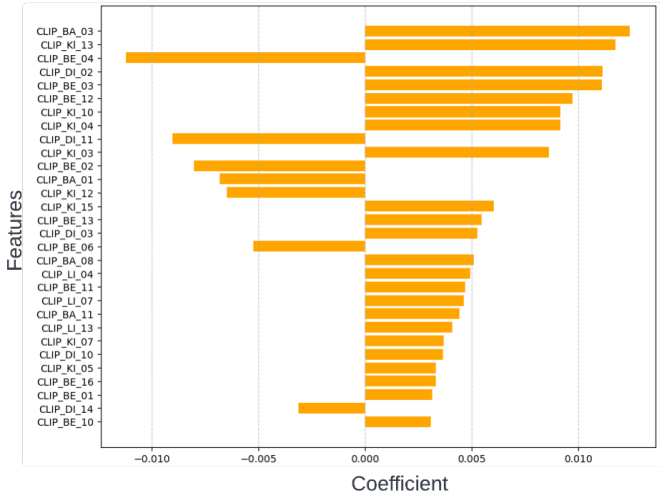


Figure 4.13: Top 30 features - apartment AVM [Ridge] - with only CLIP features

The apartment XGBoost model, with the mean difference of -0.3, is demonstrated in Figure 4.14, which shows the feature importance of the top features for the AVM. In this model, seven visual features are placed at the top 30 of the model. The top feature is related to natural light in the bedroom (BE_16). Subsequently, the flooring quality in the bedroom (BE_03), an aesthetically pleasing living room (LI_12), quality flooring and damage-free kitchen (KI_03, KI_07). Additionally, trendy fixtures in the bathroom (BA_06), and a newly renovated bedroom (BE_13) are also present at the top.

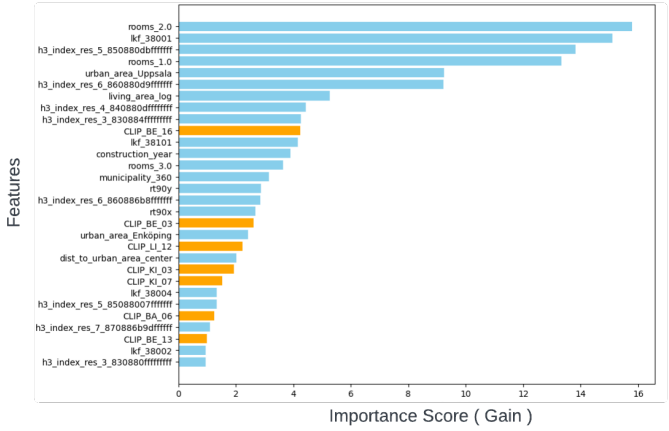


Figure 4.14: Top 30 features - apartment AVM [XGBoost] - with CLIP features

It is observed in Figure 4.15 that the top features have the strongest weights, followed by a tail of lower features. Since the top seven have already been explored, the focus will be on the next five features. These include features from the living room, kitchen, and bathroom. Specifically, they are about being damage-free in the living room (LI_07), having trendy fixtures in the bedroom (BE_06), having a spacious layout

in the kitchen (KI_02), the quality of flooring (BA_03) and the aestheticism of the bedroom (BE_12).

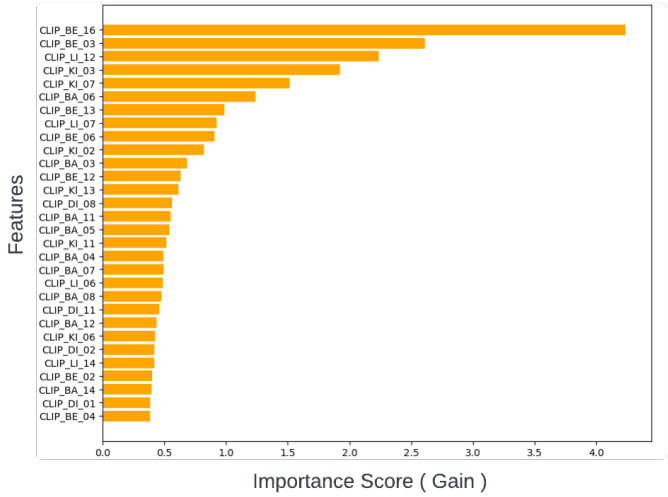


Figure 4.15: Top 30 features - apartment AVM [XGBoost] - only CLIP features

4.5.4.2 House

The focus will now shift to the house models with visual features added. With a mean difference in MAPE of -0.1, the Ridge model with the additional cluster features can be seen in Figure 4.16. The feature with the highest impact among the clusters is BE_bed, which indicates the appearance of a clear bed placement in the bedroom. The impact of no furniture in the living room and bedroom follows. The appearance of a glass shower and appliances in the bathroom also significantly impacts the model. The lower-impact features in this comparison are the placement of the TV and the fireplace in the living room, followed by a bright bedroom and the presence of an oven, which appears to be close to zero.

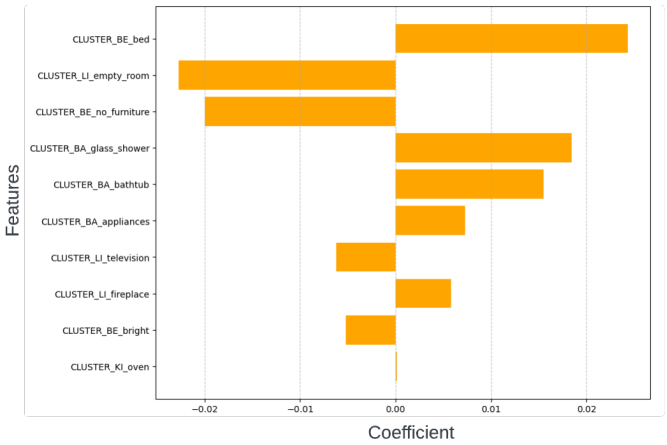


Figure 4.16: Top features - house AVM [Ridge] - only cluster features

With a mean difference of -0.6%, the focus shifts to the contribution of the CLIP features. Figure 4.17 shows a noteworthy result in that the flooring quality (BE_03, BA_03, LI_03, KI_03) is in four of the top five but for the different rooms. The additional feature, BE_06, relates to trendy fixtures.

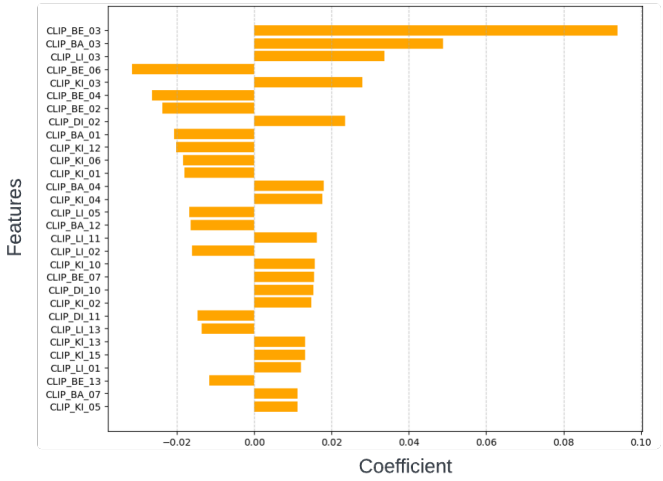


Figure 4.17: Top 30 features - house AVM [Ridge] - only CLIP features

Changing to the XGBoost model with the additional CLIP features reduced the mean difference by -0.91. Figure 4.18 shows the top feature importance of all features. As can be seen, 13 of the CLIP features were assigned such significant weight in the model that they made it into the top 30. These features highlight several aspects of the condition of the property, such as the quality of the flooring in the bedroom, bathroom, living room and kitchen (BE_03, BA_03, LI_03, KI_03). In addition, the model indicates areas free from water or mold damage in the kitchen, bathroom, and living room (KI_07, BA_07, LI_07) and features that enhance functionality and aesthetics, such as a spacious layout in the dining room (DI_02) and aesthetically pleasing characteristics in the living room (LI_04). Furthermore, it shows a seamless flow into adjacent rooms in the dining and living rooms (DI_11, LI_11), an aesthetically pleasing living room (LI_12), and recent renovations in the bedroom (BE_13).

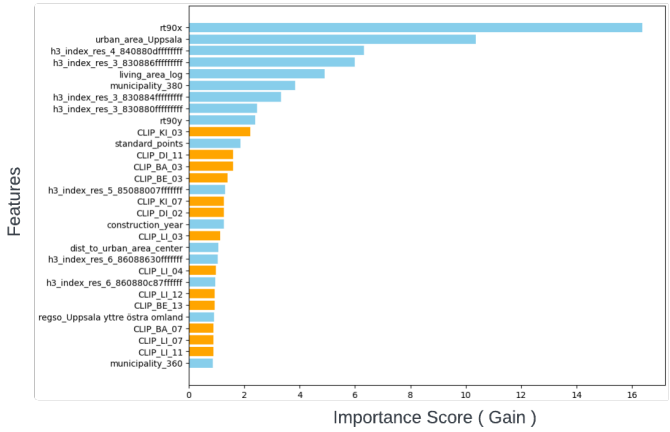


Figure 4.18: Top 30 features - house AVM [XGBoost] - with CLIP features

Finally, Figure 4.19 shows the feature importance of the XGBoost model with only CLIP features for the houses AVM. Since the first sixteen have already have been explained in the top feature figure above, the graph depicts a slow decline while maintaining high weights on the consecutive features. Notably, the continued importance of the additional features regarding flow into adjacent rooms (KL_11, BA_11, BE_11) and aesthetically pleasing bedrooms (BE_12) after trendy fixtures in the bedroom (BE_06) is present.

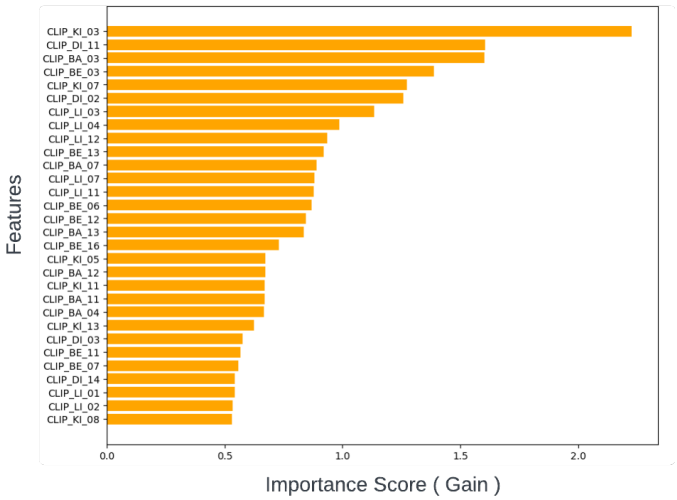


Figure 4.19: Top 30 features - house AVM [XGBoost] - only CLIP features

5

Conclusion

This chapter will summarise the findings from our study, discuss them, draw conclusions, and explore the findings' implications. Additionally, it will explore possible further work and conclude with a summary of the thesis's conclusions.

5.1 Summary of Results

The first stage of separating the interior images performed well, achieving an accuracy of 98.9%, which was in line with anticipated outcomes based on previous scores on the topic. Subsequently, the second stage of the room classification also performed well, distinguishing room types with an F1 score above 0.90% for all the room types.

Meanwhile, the AUC scores on the percentage error indicate that the pre-trained ViT model outperformed the other options in our study, including the self-supervised model trained in this thesis. The ViT showed an improved ability to distinguish desirable and undesirable features from the images compared to the other CNN-based models. Our results also indicate, in the form of an AUC score, that the kitchen is the most informative room in relation to the prediction of the price. However, using the images to predict the provided Swedish Tax Agency standard point did not achieve the desired result.

In the cluster-finding processes aimed at finding clusters connected to the architectural qualities, DINO V2 was the backbone model that resulted in recognisable clusters that remained after sampling. In terms of cluster features, the only models that improved were the Ridge models, with a slight -0.1 reduction of MAPE on apartments and houses. The visual features that improved the models were the following. In the apartment, the lack of furniture in the bedroom was the strongest indicator, followed by the glass shower. In the house cluster features, the clear placement of the bed was the strongest feature, followed by no furniture in the living room and bedroom.

In our study, using the CLIP features related to architectural qualities resulted in a significant reduction in the house model between -0.6 and -4.03. Meanwhile, the resulting reduction of the apartment models is between -0.3 and -0.59, except for the neural network model, which could not reject the null hypothesis.

In the Ridge apartment model the top CLIP features were related to the floor

quality of the bathroom, the kitchen's renovated status and an aesthetically pleasing bedroom. Sequentially, the model also rated a spacious dining room layout, and bedroom floor quality high. In the case of the XGBoost apartment model, the CLIP features were not only included but also managed to get multiple placements among the top 30 features. The most noticeable features are natural light, which ranks high, along with the quality of the bedroom and kitchen flooring, an aesthetically pleasing living room, and a damage-free kitchen.

In the Ridge house model, the top CLIP features were connected to the flooring qualities of the bedroom, bathroom, living room, and kitchen. Additionally, the feature for trendy fixtures was highlighted among the top features. Meanwhile, the XGBoost house models' visual features scored high on flooring quality, and water or mold damages and spacious layout. Notably, it achieved 13 of the top 30 features importance placements for the model.

5.2 Discussion

The self-supervised model's lower performance compared to the pre-trained ViT model is likely due to the small sample size and the minimum training period, which is probably substantially less than required to outperform the provided pre-trained model with a large, diverse dataset.

The AUC score on the percentage error also indicates that more information could be extracted from the images. It should be noted that even the highest AUC is below the desirable score of 0.8 to be considered a satisfactory divider. However, given that this is only one room to indicate a lower and higher market value, it is impressive that it is almost a sufficient separator.

The number of labels created in the initial process of this thesis was excessive, and we believe that an equally good result could have been achieved with considerably fewer labels, especially if using the pseudo-label techniques. However, in terms of quality, the best practice of cross-validation was not used, and it still had good separation, but that is a strong oversight given that we had the option.

Another comment is on the smaller sample size of the dining room images. This is probably due to the dining room being part of the kitchen or the living room and therefore not being isolated in a single image. The issue with open floor plans is also present. When the classification was on room types, in an open floor plan, images with only one room at a time were more limited.

Given this thesis's focus on feature extraction, using a labelling resource such as Amazon Mechanical Turk to extract a larger set of features separately could have been feasible. However, this study aimed to explore ViT's ability and computer vision improvements.

Disclaimer: The opinions expressed in this thesis are solely our own and are not necessarily those of Valueguard. Additionally, the AVM scores presented in this thesis are not representative of those provided by Valueguard.

5.3 Contributions

Our main contribution in this thesis is our exploration of the importance of visual features related to market value. To our knowledge, no similar studies have used these new transformer techniques. The use of CLIP to extract the features straightforwardly highlights that the images contain more information and that other studies can continue to explore this topic.

This thesis also tried using self-supervised models. However, these models did not show promising results possibly due to limited training in the form of epochs and data size. Consequentially, we believe that this is still an area worth exploring, especially given the findings of the qualities with the CLIP model.

5.4 Limitations of the Study

In this thesis, multiple limitations have affected the results. Time and computing power were the main ones, and the highest-scoring AUC on the ViT models was only obtained at the end of the thesis. This also includes the continuous improvement of the AVM models, which were time-consuming in the cross-validation to find better hyper-parameters, and we believe that with additional attempts, we could improve it further.

Furthermore, this study initially aimed to test how well our findings generalised in other regions. However, due to time constrains these investigations are now left for further studies.

There might also be underrepresented features or house types that will not scale for the region we analyse or the country as a whole. For the excluded analytical properties and summer cottages, the applicability of our findings to these house types, which may have different market dynamics and value determinants, is limited.

Another limitation was that the study could have benefited from being narrower. Focusing on a single house type would have provided fewer AVM models to train and allowed more focus on fine-tuning the models and exploring additional techniques, resulting in a more in-depth and organised study. The choice of both house types was initially made to allow the use of standard points that would be unique to our study.

5.5 Practical Implications

This study's findings show an increase in the ability and performance of using visual aspects in the AVM process, thus reducing the uncertainty in the assessment process. Additionally, we believe it can substantially improve the comps selection process for brokers by allowing them to compare properties along the scores in a more meaningful way that can take the architectural qualities into account.

5.5.1 Recommendations for Future Research

Given the success in identifying features related to market value, we believe that the extraction of visual features will continue to be an intriguing research area, especially with the current research on quantifying guidelines for what is considered high and low architectural qualities, which will be a valuable addition to the current literature.

Firstly, we encourage further work to verify our findings and explore other regions to see if these findings generalise across the country, given our limited study area. For example, such a study could explore regions like Åre, a famous skiing region with mostly ski apartments, and other smaller cities in the centre of Sweden, such as Mora.

Secondly, given the kitchen's relatively high AUC score, we suggest a targeted study within a more specific area. This could be done in combination with segmentation to divide the room components as wall, ceiling and floor into classes and compare them separately. We explored this slightly in our study, but the Segment Anything Model (SAM) in combination with the Self-Distillation with No Labels (DINO) model in Label Studio makes this a reasonable task for a study of a similar size to ours [117].

Thirdly, future research could be more targeted to a specific customer group, investigating only a specific type of home with a typical owner. The theory suggests that different customer groups have different requirements in terms of beauty and function. This could be explored by comparing these categories of homes.

Lastly, another possible study could combine NLP feature extraction from the ad text connected to the ad description of the property. This could then be used as the desirable and undesirable target value. This approach could be applied in the form of matching similar to CLIP or by using the tone of the description, as with other studies where the text was used to indicate desirable and undesirable traits.

5.5.2 Conclusive Summary

This research was conducted as a master's thesis in the field of Data Science and AI. It explored the potential of incorporating visual features related to architectural qualities extracted from interior property images to improve automated valuation models in Sweden. This has been studied by employing computer vision techniques, particularly Vision Transformers, and combining them with metadata such as location and living area to measure the reduced uncertainty in the predictions.

Despite the challenging nature of quantifying architectural qualities, this thesis has shown promising results that align with previous research. The extracted visual features, notably those derived from Contrastive Language-Image Pre-Training (CLIP), are essential in identifying architectural patterns and characteristics related to market value, offering practical insights for real estate professionals.

Bibliography

[1] F. Brunes, *Värdering av småhus och bostadsrätter*, Swedish, 1st ed. Lund, Sweden: Studentlitteratur AB, 2018, ISBN: 9789144118390.

[2] H. Sharma, H. Harsora, and B. Ogunleye, "An optimal house price prediction algorithm: Xgboost," *Analytics*, vol. 3, no. 1, pp. 30–45, Jan. 2024, ISSN: 2813-2203. DOI: 10.3390/analytics3010003. [Online]. Available: http://dx.doi.org/10.3390/analytics3010003.

[3] J. Kintzel, "Price Prediction and Computer Vision in the Real Estate Marketplace," M.S. thesis, Harvard Extension School, 2019. [Online]. Available: https://nrs.harvard.edu/URN-3:HUL.INSTREPOS:37365260.

[4] O. Poursaeed, T. Matera, and S. Belongie, "Vision-based real estate price estimation," *Machine Vision and Applications*, vol. 29, pp. 667–676, 2018. DOI: 10.1007/s00138-018-0922-2. [Online]. Available: https://doi.org/10.1007/s00138-018-0922-2.

[5] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, ISSN: 1558-2256. DOI: 10.1109/5.726791.

[6] D. V. Nieto, L. Celona, and C. Fernandez-Labrador, *Understanding aesthetics with language: A photo critique dataset for aesthetic assessment*, 2022. arXiv: 2206.08614 [cs.CV].

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *CoRR*, vol. abs/2010.11929, 2020. arXiv: 2010.11929. [Online]. Available: https://arxiv.org/abs/2010.11929.

[8] M. Caron, H. Touvron, I. Misra, *et al.*, "Emerging properties in self-supervised vision transformers," *CoRR*, vol. abs/2104.14294, 2021. arXiv: 2104.14294. [Online]. Available: https://arxiv.org/abs/2104.14294.

[9] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. E. Hinton, "Big self-supervised models are strong semi-supervised learners," *CoRR*, vol. abs/2006.10029, 2020. arXiv: 2006.10029. [Online]. Available: https://arxiv.org/abs/2006.10029.

[10] A. Radford, J. W. Kim, C. Hallacy, *et al.*, "Learning transferable visual models from natural language supervision," *arXiv preprint arXiv:2103.00020*, 2021.

[11] Z. Kostic and A. Jevremovic, "What Image Features Boost Housing Market Predictions?" *IEEE Transactions on Multimedia*, vol. 22, no. 7, pp. 1904–1916, 2020. DOI: 10.1109/TMM.2020.2966890.

[12] M. Al-Omari, "The role of reliable land valuations in land management and land administration systems efficiency," in *Proceedings of the FIG Working Week*, 2008.

[13] Q. You, R. Pang, L. Cao, and J. Luo, "Image-based appraisal of real estate properties," *IEEE Transactions on Multimedia*, vol. 19, no. 12, pp. 2751–2759, 2017. DOI: 10.1109/TMM.2017.2710804.

[14] *Villor - prisutveckling i riket*, Accessed: 2024-05-01, Svensk Mäklarstatistik, 2024. [Online]. Available: https://www.maklarstatistik.se/omrade/riket/#/villor/arshistorik-prisutveckling.

[15] *Bostadsrätter - prisutveckling i riket*, Accessed: 2024-05-01, Svensk Mäklarstatistik, 2024. [Online]. Available: https://www.maklarstatistik.se/omrade/riket/#/bostadsratter.

[16] D. Ashok, "Estimating the price of real estate properties with the help of online images," *ECS Transactions*, vol. 107, no. 1, p. 16 933, Apr. 2022. DOI: 10.1149/10701.16933ecst. [Online]. Available: https://dx.doi.org/10.1149/10701.16933ecst.

[17] S. Adcock, "Managing risks in property exposure via valuations/appraisal assessments," in *Proceedings of the FIG XXII International Congress*, Washington, D.C., USA, Apr. 2002, pp. 1–14.

[18] Danske Bank, *Värdera bostad hur går en värdering av bostad till?* https://danskebank.se/privat/produkter/bolan/guider/vardera-bostad, Accessed: 2024-05-02, 2024.

[19] Valueguard Index Sweden AB, *Short methodology and data description*, Revised at 2023-08-21, Aug. 2023. [Online]. Available: https://valueguard.se/static/media/vg_MethodAndDataDesc_2023aug.23592fcf.pdf.

[20] C. Lee and K.-H. Park, "Using photographs and metadata to estimate house prices in south korea," *Data Technologies and Applications*, vol. 55, no. 2, pp. 280–292, 2021. DOI: 10.1108/DTA-05-2020-0111.

[21] Booli, *Fördjupning: Så funkar boolis värderingar*, https://www.booli.se/kunskap/fordjupning-sa-funkar-boolis-varderingar/, Accessed: 2024-02-23, 2022.

[22] N. N. Ghosalkar and S. N. Dhage, "Real estate value prediction using linear regression," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 2018, pp. 1–5. DOI: 10.1109/ICCUBEA.2018.8697639.

[23] *Xgboost documentation xgboost 2.0.3*, https://xgboost.readthedocs.io/en/stable/, Accessed: 2024-05-02.

[24] Zillow, *Building the neural zestimate*, https://www.zillow.com/tech/building-the-neural-zestimate/, Accessed: 2024-04-22, 2022.

[25] Vitruvius, *The Ten Books on Architecture*, English, trans. by M. H. Morgan. New York: Dover Publications, 1960, vol. 1, p. 368, ISBN: 978-0486206455.

[26] O. Nylander and K. Forshed, *Bostadens omätbara värden*.

[27] P. Madhavan and M. Liechty, "Toward an understanding of real estate homebuyer internet search behavior: An application of ocular tracking technology," *Journal of Real Estate Research*, vol. 34, Jun. 2011. DOI: 10.1080/10835547.2012.12091333.

[28] C. Björk, P. Kallstenius, and L. Reppen, *Så byggdes husen 1880-2020*. Svensk Byggtjänst, 2021, p. 172, ISBN: 9789179170943.

[29] C. Björk, L. Nordling, and L. Reppen, *Så byggdes staden*. Svensk Byggtjänst, 2023, p. 229, ISBN: 9789179171797.

[30] G. Bergström, C. Björk, and L. Reppen, *Tidstypiska kök & bad 1880-2000*. Svensk Byggtjänst, 2020, p. 264, ISBN: 9789179170370.

[31] Fastighetsnytt, *Vad är arkitektonisk kvalitet?* Opinion piece on architectural quality, Fastighetsnytt. [Online]. Available: https://www.fastighetsnytt.se/opinion/kronika/vad-ar-arkitektonisk-kvalitet/.

[32] B. Forshed, *Kan vi objektivt mäta bostäders kvalitet?* Article on measuring the quality of housing objectively, Brunnberg Forshed Arkitektkontor. [Online]. Available: https://www.brunnbergoforshed.se/kan-vi-objektivt-mata-bostaders-kvalitet/.

[33] X. Wang, Y. Takada, Y. Kado, and T. Yamasaki, "Predicting the Attractiveness of Real-Estate Images by Pairwise Comparison using Deep Learning," in *2019 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, IEEE, 2019, pp. 84–89. DOI: 10.1109/ICMEW.2019.0-106.

[34] P. D. Kumkar, "Image-based real estate appraisal using cnns and ensemble learning," Master's Project, San Jose State University, San Jose, California, Spring 2021. DOI: 10.31979/etd.km4q-65hg. [Online]. Available: https://scholarworks.sjsu.edu/etd_projects/1015/.

[35] J. H. Bappy, J. R. Barr, N. Srinivasan, and A. K. Roy-Chowdhury, "Real Estate Image Classification," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, 2017, pp. 373–381. DOI: 10.1109/WACV.2017.48.

[36] S. Elnagar and M. A. Thomas, "Real estate image-based appraisal using mask region based convolutional networks," 2019. [Online]. Available: https://www.researchgate.net/profile/Samaa-Elnagar/publication/346954641_Image-Based_Appraisal_for_Real_Estate_Using_Mask_Region_Convolutional_Networks_Completed_Research_Full_Paper/links/5fd3a4d1299bf1408800b012/Image-Based-Appraisal-for-Real-Estate-Using-Mask-Region-Convolutional-Networks-Completed-Research-Full-Paper.pdf.

[37] S. Thaler and D. Koch, "Real estate pictures: The role of furniture preferences in subjective valuation," *Journal of Housing Research*, vol. 32, no. 2, pp. 180–203, 2023. DOI: 10.1080/10527001.2023.2168585. eprint: https://doi.org/10.1080/10527001.2023.2168585. [Online]. Available: https://doi.org/10.1080/10527001.2023.2168585.

[38] M. D. Nadai and B. Lepri, "The economic value of neighborhoods: Predicting real estate prices from the urban environment," *CoRR*, vol. abs/1808.02547, 2018. arXiv: 1808.02547. [Online]. Available: http://arxiv.org/abs/1808.02547.

[39] S. Law, B. Paige, and C. Russell, "Take a look around: Using street view and satellite images to estimate house prices," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 5, Sep. 2019, ISSN: 2157-6904. DOI: 10.1145/3342240. [Online]. Available: https://doi.org/10.1145/3342240.

[40] Y. Zhang and R. Dong, "Impacts of street-visible greenery on housing prices: Evidence from a hedonic price model and a massive street view image dataset in beijing," *ISPRS International Journal of Geo-Information*, vol. 7, no. 3, 2018, ISSN: 2220-9964. DOI: 10.3390/ijgi7030104. [Online]. Available: https://www.mdpi.com/2220-9964/7/3/104.

[41] *Street view api documentation*, Google Maps Platform, Google. [Online]. Available: https://developers.google.com/maps/documentation/streetview/overview.

[42] *Maps static api documentation*, Google Maps Platform, Google. [Online]. Available: https://developers.google.com/maps/documentation/maps-static/start.

[43] A. Nouriani and L. Lemke, "Vision-based housing price estimation using interior, exterior satellite images," *Intelligent Systems with Applications*, vol. 14, p. 200081, 2022, ISSN: 2667-3053. DOI: https://doi.org/10.1016/j.iswa.2022.200081. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667305322000217.

[44] C. Kmen, "Use of pictures from social media to assess the local attractivity as an indicator for real estate value assessment," Diploma Thesis, Technische Universität Wien, Vienna, 2017, p. 80. DOI: 10.34726/hss.2017.38166. [Online]. Available: https://repositum.tuwien.at/handle/20.500.12708/5764.

[45] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[46] J. Perktold, S. Seabold, J. Taylor, and statsmodels-developers, *Ols — statsmodels 0.15.0 documentation*, https://www.statsmodels.org/dev/generated/statsmodels.regression.linear_model.OLS.html, Accessed: 2024-05-02, 2024.

[47] A. Sherstinsky, "Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network," *Physica D: Nonlinear Phenomena*, vol. 404, p. 132306, Mar. 2020, ISSN: 0167-2789. DOI: 10.1016/j.physd.2019.132306. [Online]. Available: http://dx.doi.org/10.1016/j.physd.2019.132306.

[48] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," vol. 25, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds., 2012. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[49] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, 2015. arXiv: 1409.1556 [cs.CV].

[50] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, Ieee, 2009, pp. 248–255.

[52] A. Torralba *et al.*, *Mit indoor scene recognition*, https://web.mit.edu/torralba/www/indoor.html, Accessed: 2024-05-02, 2024.

[53] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[54] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, *Attention is all you need*, 2023. arXiv: 1706.03762 [cs.CL].

[55] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. arXiv: 1810.04805. [Online]. Available: http://arxiv.org/abs/1810.04805.

[56] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," *OpenAI*, 2018, Accessed: 2024-05-29. [Online]. Available: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

[57] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, *An image is worth 16x16 words: Transformers for image recognition at scale*, 2021. arXiv: 2010.11929 [cs.CV].

[58] M. Fitzpatrick, V. Gujral, A. Kapoor, and A. Wolkomir. "Generative ai can change real estate, but the industry must change to reap the benefits." (2023), [Online]. Available: https://www.mckinsey.com/industries/real-estate/our-insights/generative-ai-can-change-real-estate-but-the-industry-must-change-to-reap-the-benefits (visited on 05/16/2024).

[59] V. I. S. AB, *Valueguard index sweden ab*, https://valueguard.se/, Accessed: 09 May 2024.

[60] OpenStreetMap contributors, *Planet dump retrieved from https://planet.osm.org*, https://www.openstreetmap.org, 2017.

[61] *Python: A dynamic, open source programming language*, https://www.python.org/, Accessed: 2024-04-24, 2024.

[62] A. Paszke, S. Gross, F. Massa, *et al.*, *PyTorch – An open source machine learning framework that accelerates the path from research prototyping to production deployment*, https://pytorch.org/, Accessed: 2024-04-24, 2024.

[63] J. V. den Bossche *et al.*, *GeoPandas: Python tools for geographic data*, http://geopandas.org, Accessed: 2024-04-24, 2024.

[64] W. McKinney *et al.*, *pandas: A foundational python library for data analysis and statistics*, https://pandas.pydata.org/, Accessed: 2024-04-24, 2024.

[65] P. Jupyter *et al.*, *Jupyter Notebook: An open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text*, https://jupyter.org/, Accessed: 2024-04-24, 2024.

[66] P. Jupyter, *Jupyter Data Science Notebook: A docker image for data science with jupyter notebook*, https://hub.docker.com/r/jupyter/datascience-notebook, Accessed: 2024-04-24, 2024.

[67] HumanSignal, Inc., *Label studio*, https://labelstud.io/, Accessed: 2024-02-22.

[68] Databricks, *Mlflow: An open source platform for the machine learning lifecycle*, https://mlflow.org/, Accessed: 2024-04-25, 2023.

[69] R. Merritt, *Why gpus are great for ai*, Accessed: 2024-05-15, NVIDIA, Dec. 2023. [Online]. Available: https://blogs.nvidia.com/blog/why-gpus-are-great-for-ai/.

[70] L. Alzubaidi, J. Zhang, A. J. Humaidi, *et al.*, "Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, pp. 1–74, 2021. DOI: 10.1186/s40537-021-00444-8. [Online]. Available: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00444-8.

[71] *Valuegard cma tool*, Online, Accessed: [Date you accessed the site], Valuegard, 2024. [Online]. Available: https://valueguard.se/serviceintag.

[72] *Valuegard avm tool*, Online, Accessed: [Date you accessed the site], Valuegard, 2024. [Online]. Available: https://valueguard.se/serviceavm.

[73] V. I. S. AB, *Method and data description: Valueguard index, august 2023*, https://valueguard.se/static/media/vg_MethodAndDataDesc_2023aug.23592fcf.pdf, Accessed: 09 May 2024.

[74] Unsplash, *Unsplash license*, https://unsplash.com/license, Accessed: 2024-05-22, 2024.

[75] S. Centralbyrån, *Statistics sweden*, https://www.scb.se, Accessed: 2024-04-24, 2024.

[76] Statistics Sweden, *Deso - demografiska statistikområden*, Accessed: 2024-04-25, Statistics Sweden, 2024. [Online]. Available: https://www.scb.se/hitta-statistik/regional-statistik-och-kartor/regionala-indelningar/deso---demografiska-statistikomraden/.

[77] Statistics Sweden, *Regso - regionala statistikområden*, Accessed: 2024-04-25, Statistics Sweden, 2024. [Online]. Available: https://www.scb.se/hitta-statistik/regional-statistik-och-kartor/regionala-indelningar/regso---regionala-statistikomraden/.

[78] Statistics Sweden, *Öppna geodata: Tätorter*, Accessed: 2024-04-25, Statistics Sweden, 2024. [Online]. Available: https://www.scb.se/vara-tjanster/oppna-data/oppna-geodata/tatorter/.

[79] Statistics Sweden, *Län och kommuner*, Accessed: 2024-04-25, Statistics Sweden, 2024. [Online]. Available: https://www.scb.se/hitta-statistik/regional-statistik-och-kartor/regionala-indelningar/lan-och-kommuner/.

[80] U. T. Inc., *H3: A hexagonal hierarchical geospatial indexing system*, https://github.com/uber/h3, Accessed: 2024-04-24, 2024.

[81] Google Developers, *Normalization: Transform data*, https://developers.google.com/machine-learning/data-prep/transform/normalization, Accessed: 2024-05-16, May 2024.

[82] S. Mäklarstatistik, *Svensk mäklarstatistik - riket, bostadsrätter by room types*, https://www.maklarstatistik.se/omrade/riket/#/bostadsratter/48m-rum, Accessed: 09 May 2024.

[83] S. S. (SCB), *Statistical review of municipal taxes and fees 2021*, https://www.scb.se/contentassets/13ec5841d80045498d960d456e87ea78/lkf2021_2020-06-15.pdf, Accessed: 09 May 2024.

[84] Skatteverket, *Innehållet i fastighetsdeklarationen för småhus*, https://www.skatteverket.se/privat/fastigheterochbostad/fastighetstaxering/deklarerasmahus/innehalletifastighetsdeklarationen.4.515a6be615c637b9aa41532d.html, Accessed: 09 May 2024.

[85] scikit-learn developers, *Sklearn.feature_selection.selectfrommodel — scikit-learn 1.2.2 documentation*, Accessed: 2024-04-25, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectFromModel.html.

[86] scikit-learn developers, *Sklearn.pipeline.pipeline - scikit-learn 1.1.3 documentation*, scikit-learn documentation, Accessed: 2024-04-25, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.pipeline.Pipeline.html.

[87] scikit-learn developers, *Sklearn.preprocessing.standardscaler - scikit-learn 1.1.3 documentation*, scikit-learn documentation, Accessed: 2024-04-25, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html.

[88] scikit-learn developers, *Sklearn.preprocessing.onehotencoder - scikit-learn 1.1.3 documentation*, scikit-learn documentation, Accessed: 2024-04-25, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OneHotEncoder.html.

[89] T. Documentation, *Vision transformer (vit) model - vit_b_16*, Accessed: 2024-05-21, 2023. [Online]. Available: https://pytorch.org/vision/main/models/generated/torchvision.models.vit_b_16.html#torchvision.models.vit_b_16.

[90] T. Documentation, *Vgg-13 model - vgg_13*, Accessed: 2024-05-21, 2023. [Online]. Available: https://pytorch.org/vision/main/models/generated/torchvision.models.vgg13.html#torchvision.models.vgg13.

[91] T. Documentation, *Resnet-50 model - resnet_50*, Accessed: 2024-05-21, 2023. [Online]. Available: https://pytorch.org/vision/main/models/generated/torchvision.models.resnet50.html.

[92] *Torchvision.transforms torchvision 0.9.0 documentation*, https://pytorch.org/vision/0.9/transforms.html, Accessed: 2023-05-23.

[93] P.-I. Contributors, *Earlystopping handler*, https://pytorch.org/ignite/generated/ignite.handlers.early_stopping.EarlyStopping.html, Accessed: 2024-05-21, 2023.

[94] J. Brownlee, *Batch normalization for training of deep neural networks*, https://machinelearningmastery.com/batch-normalization-for-training-of-deep-neural-networks/, Accessed: 2024-05-16, 2024.

[95] TensorFlow, *Transfer learning & fine-tuning*, https://www.tensorflow.org/guide/keras/transfer_learning, Accessed: 2024-05-21, 2023.

[96] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," *arXiv preprint arXiv:2002.05709*, 2020.

[97] G. Research, *Simclr: A simple framework for contrastive learning of visual representations*, Accessed: 2024-04-26, 2020. [Online]. Available: https://github.com/google-research/simclr.

[98] M. Caron, H. Touvron, I. Misra, *et al.*, "Emerging properties in self-supervised vision transformers," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[99] F. Research, *Pytorch code for vision transformers training with the self-supervised learning method dino*, Accessed: 2024-04-26, 2021. [Online]. Available: https://github.com/facebookresearch/dino.

[100] D. Bergmann, "What is zero-shot learning?," 2024, Accessed: 2024-05-28. [Online]. Available: https://www.ibm.com/topics/zero-shot-learning.

[101] Evidently AI, *Explain roc curve*, https://www.evidentlyai.com/classification-metrics/explain-roc-curve, Accessed: 2024-05-16, 2024.

[102] *Roc_auc_score - scikit-learn 1.2.2 documentation*, https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html, Accessed: date-of-access, scikit-learn, 2024.

[103] M. Oquab, T. Darcet, T. Moutakanni, *et al.*, *Dinov2: Learning robust visual features without supervision*, 2023.

[104] T. Darcet, M. Oquab, J. Mairal, and P. Bojanowski, *Vision transformers need registers*, 2023.

[105] S.-L. Developers, *Davies-bouldin score*, https://scikit-learn.org/stable/modules/generated/sklearn.metrics.davies_bouldin_score.html, Accessed: 2024-05-21, 2023.

[106] OpenAI, *Clip image*, https://github.com/openai/CLIP/blob/main/CLIP.png, Accessed: 2024-05-21, 2024.

[107] T. Chen and C. Guestrin, *Xgboost: A scalable tree boosting system*, Accessed: 2024-05-23, 2016. [Online]. Available: https://xgboost.readthedocs.io/en/stable/.

[108] scikit-learn, *Metrics and scoring: Quantifying the quality of predictions scikit-learn 1.5.0 documentation*, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/model_evaluation.html#mean-squared-error.

[109] *Metrics and scoring: Quantifying the quality of predictions scikit-learn 1.5.0 documentation*, Accessed: 2024-05-23, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/model_evaluation.html#mean-absolute-error.

[110] *Metrics and scoring: Quantifying the quality of predictions scikit-learn 1.5.0 documentation*, Accessed: 2024-05-23, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/model_evaluation.html#mean-absolute-percentage-error.

[111] *Metrics and scoring: Quantifying the quality of predictions scikit-learn 1.5.0 documentation*, Accessed: 2024-05-23, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/model_evaluation.html#r2-score.

[112] *Metrics and scoring: Quantifying the quality of predictions scikit-learn 1.5.0 documentation*, Accessed: 2024-05-23, 2024. [Online]. Available: https://

scikit-learn.org/stable/modules/model_evaluation.html#median-absolute-error.

[113] X. Documentation, *Xgboost.booster.get_score - xgboost 1.0.0 documentation*, Accessed: 2024-05-23, 2024. [Online]. Available: https://xgboost.readthedocs.io/en/release_1.0.0/python/python_api.html#xgboost.Booster.get_score.

[114] P. Documentation, *Torch.nn.crossentropyloss*, Accessed: 2024-05-23, 2024. [Online]. Available: https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html.

[115] M. Krzywinski and N. Altman, "Significance, p values and t-tests," *Nature Methods*, vol. 10, pp. 1041–1042, 2013. DOI: 10.1038/nmeth.2698. [Online]. Available: https://www.nature.com/articles/nmeth.2698.

[116] M. Aickin and H. Gensler, "Adjusting for multiple testing when reporting research results: The bonferroni vs holm methods," *American Journal of Public Health*, vol. 86, no. 5, pp. 726–728, May 1996. DOI: 10.2105/ajph.86.5.726.

[117] *Using text prompts for image annotation with grounding dino and label studio*, Accessed: 2024-05-15, Label Studio, 2024. [Online]. Available: https://labelstud.io/blog/using-text-prompts-for-image-annotation-with-grounding-dino-and-label-studio/.

A

CLIP features

Table A.1: CLIP features - bedroom

Architectural Quality	**Room Type**	**Code**	**Positive (Pos)**	**Negative (Neg)**
Stability	Bedroom	BE_01	Bedroom shows the walls and ceiling in mint condition	Bedroom with visible damages and cracks on walls
Utility	Bedroom	BE_02	Bedroom with an open, spacious layout enriching the functionality	Cramped Bedroom with bad layout
Stability	Bedroom	BE_03	Bedroom features high-quality flooring	Bedroom with damaged flooring
Beauty	Bedroom	BE_04	Aesthetically pleasing Bedroom	Outdated and ugly Bedroom
Utility	Bedroom	BE_05	Bedroom includes multiple useful built-ins	Bedroom without built-ins
Beauty	Bedroom	BE_06	Bedroom with trendy fixtures	Bedroom with unfashionable fixtures
Stability	Bedroom	BE_07	Bedroom free from water damage or mold	Bedroom with water damage or mold
Utility	Bedroom	BE_08	Well-insulated Bedroom against external noise	Poorly insulated Bedroom room against external noise
Beauty	Bedroom	BE_09	Image of a Bedroom	Not an image of Bedroom
Beauty	Bedroom	BE_10	Bedroom showcases unique architectural details	Bedroom showcases regular architectural details
Utility	Bedroom	BE_11	Bedroom with a seamlessly flows into adjacent spaces	Cramped Bedroom with bad layout
Beauty	Bedroom	BE_12	Aesthetically pleasing Bedroom	Displeasing Bedroom
Stability	Bedroom	BE_13	Newly renovated Bedroom	Bedroom in need of renovation
Beauty	Bedroom	BE_14	Bedroom with big windows	Bedroom with small or no windows
Beauty	Bedroom	BE_15	Bedroom with a lot of natural light	Bedroom with only artificial light

Table A.2: CLIP features - bathroom

Architectural Quality	**Room Type**	**Code**	**Positive (Pos)**	**Negative (Neg)**
Stability	Bathroom	BA_01	Bathroom shows the walls and ceiling in mint condition	Bathroom with visible damages and cracks on walls
Utility	Bathroom	BA_02	Bathroom with an open, spacious layout enriching the functionality	Cramped Bathroom with bad layout
Stability	Bathroom	BA_03	Bathroom features high-quality flooring	Bathroom with damaged flooring
Beauty	Bathroom	BA_04	Aesthetically pleasing Bathroom	Outdated and ugly Bathroom
Utility	Bathroom	BA_05	Bathroom includes multiple useful built-ins	Bathroom without built-ins
Beauty	Bathroom	BA_06	Bathroom with trendy fixtures	Bathroom with unfashionable fixtures
Stability	Bathroom	BA_07	Bathroom free from water damage or mold	Bathroom with water damage or mold
Utility	Bathroom	BA_08	Well-insulated Bathroom against external noise	Poorly insulated Bathroom room against external noise
Beauty	Bathroom	BA_09	Image of a Bathroom	Not an image of Bathroom
Beauty	Bathroom	BA_10	Bathroom showcases unique architectural details	Bathroom showcases regular architectural details
Utility	Bathroom	BA_11	Bathroom with a seamlessly flows into adjacent spaces	Cramped Bathroom with bad layout
Beauty	Bathroom	BA_12	Aesthetically pleasing Bathroom	Displeasing Bathroom
Stability	Bathroom	BA_13	Newly renovated Bathroom	Bathroom in need of renovation
Beauty	Bathroom	BA_14	Bathroom with big windows	Bathroom with small or no windows
Beauty	Bathroom	BA_15	Bathroom with a lot of natural light	Bathroom with only artificial light

Table A.3: CLIP features - kitchen

Architectural Quality	**Room Type**	**Code**	**Positive (Pos)**	**Negative (Neg)**
Stability	Kitchen	KI_01	Kitchen shows the walls and ceiling in mint condition	Kitchen with visible damages and cracks on walls
Utility	Kitchen	KI_02	Kitchen with an open, spacious layout enriching the functionality	Cramped Kitchen with bad layout
Stability	Kitchen	KI_03	Kitchen features high-quality flooring	Kitchen with damaged flooring
Beauty	Kitchen	KI_04	Aesthetically pleasing Kitchen	Outdated and ugly Kitchen
Utility	Kitchen	KI_05	Kitchen includes multiple useful built-ins	Kitchen without built-ins
Beauty	Kitchen	KI_06	Kitchen with trendy fixtures	Kitchen with unfashionable fixtures
Stability	Kitchen	KI_07	Kitchen free from water damage or mold	Kitchen with water damage or mold
Utility	Kitchen	KI_08	Well-insulated Kitchen against external noise	Poorly insulated Kitchen room against external noise
Beauty	Kitchen	KI_09	Image of a Kitchen	Not an image of Kitchen
Beauty	Kitchen	KI_10	Kitchen showcases unique architectural details	Kitchen showcases regular architectural details
Utility	Kitchen	KI_11	Kitchen with a seamlessly flows into adjacent spaces	Cramped Kitchen with bad layout
Beauty	Kitchen	KI_12	Aesthetically pleasing Kitchen	Displeasing Kitchen
Stability	Kitchen	KI_13	Newly renovated Kitchen	Kitchen in need of renovation
Beauty	Kitchen	KI_14	Kitchen with big windows	Kitchen with small or no windows
Beauty	Kitchen	KI_15	Kitchen with a lot of natural light	Kitchen with only artificial light

Table A.4: CLIP features - living room

Architectural Quality	**Room Type**	**Code**	**Positive (Pos)**	**Negative (Neg)**
Stability	Living Room	LI_01	Living Room shows the walls and ceiling in mint condition	Living Room with visible damages and cracks on walls
Utility	Living Room	LI_02	Living Room with an open, spacious layout enriching the functionality	Cramped Living Room with bad layout
Stability	Living Room	LI_03	Living Room features high-quality flooring	Living Room with damaged flooring
Beauty	Living Room	LI_04	Aesthetically pleasing Living Room	Outdated and ugly Living Room
Utility	Living Room	LI_05	Living Room includes multiple useful built-ins	Living Room without built-ins
Beauty	Living Room	LI_06	Living Room with trendy fixtures	Living Room with unfashionable fixtures
Stability	Living Room	LI_07	Living Room free from water damage or mold	Living Room with water damage or mold
Utility	Living Room	LI_08	Well-insulated Living Room against external noise	Poorly insulated Living Room room against external noise
Beauty	Living Room	LI_09	Image of a Living Room	Not an image of Living Room
Beauty	Living Room	LI_10	Living Room showcases unique architectural details	Living Room showcases regular architectural details
Utility	Living Room	LI_11	Living Room with a seamlessly flows into adjacent spaces	Cramped Living Room with bad layout
Beauty	Living Room	LI_12	Aesthetically pleasing Living Room	Displeasing Living Room
Stability	Living Room	LI_13	Newly renovated Living Room	Living Room in need of renovation
Beauty	Living Room	LI_14	Living Room with big windows	Living Room with small or no windows
Beauty	Living Room	LI_15	Living Room with a lot of natural light	Living Room with only artificial light

Table A.5: CLIP features - dining room

Architectural Quality	**Room Type**	**Code**	**Positive (Pos)**	**Negative (Neg)**
Stability	Dining Room	DI_01	Dining Room shows the walls and ceiling in mint condition	Dining Room with visible damages and cracks on walls
Utility	Dining Room	DI_02	Dining Room with an open, spacious layout enriching the functionality	Cramped Dining Room with bad layout
Stability	Dining Room	DI_03	Dining Room features high-quality flooring	Dining Room with damaged flooring
Beauty	Dining Room	DI_04	Aesthetically pleasing Dining Room	Outdated and ugly Dining Room
Utility	Dining Room	DI_05	Dining Room includes multiple useful built-ins	Dining Room without built-ins
Beauty	Dining Room	DI_06	Dining Room with trendy fixtures	Dining Room with unfashionable fixtures
Stability	Dining Room	DI_07	Dining Room free from water damage or mold	Dining Room with water damage or mold
Utility	Dining Room	DI_08	Well-insulated Dining Room against external noise	Poorly insulated Dining Room room against external noise
Beauty	Dining Room	DI_09	Image of a Dining Room	Not an image of Dining Room
Beauty	Dining Room	DI_10	Dining Room showcases unique architectural details	Dining Room showcases regular architectural details
Utility	Dining Room	DI_11	Dining Room with a seamlessly flows into adjacent spaces	Cramped Dining Room with bad layout
Beauty	Dining Room	DI_12	Aesthetically pleasing Dining Room	Displeasing Dining Room
Stability	Dining Room	DI_13	Newly renovated Dining Room	Dining Room in need of renovation
Beauty	Dining Room	DI_14	Dining Room with big windows	Dining Room with small or no windows
Beauty	Dining Room	DI_15	Dining Room with a lot of natural light	Dining Room with only artificial light

B

Neural Network model architectures

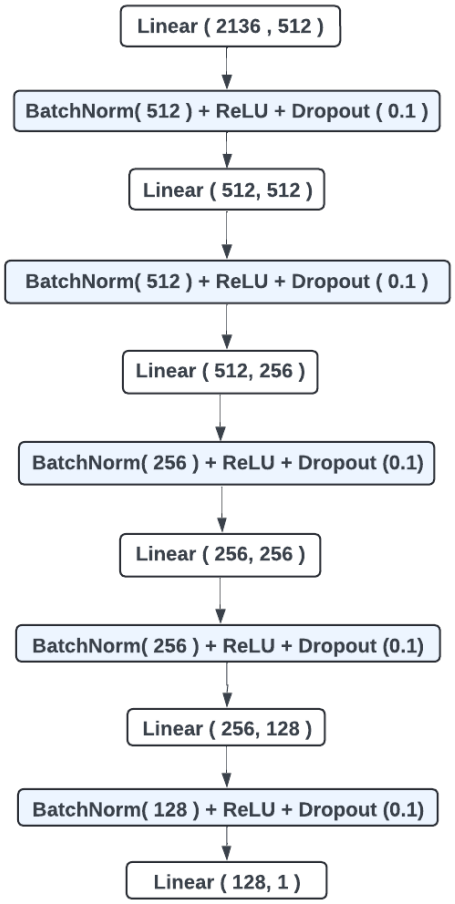


Figure B.1: Neural Network model structure for apartment AVM

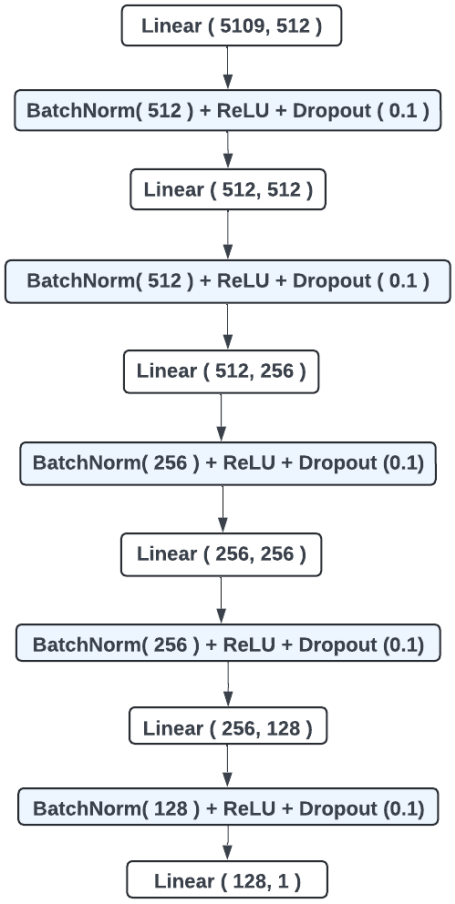


Figure B.2: Neural Network model structure for house AVM

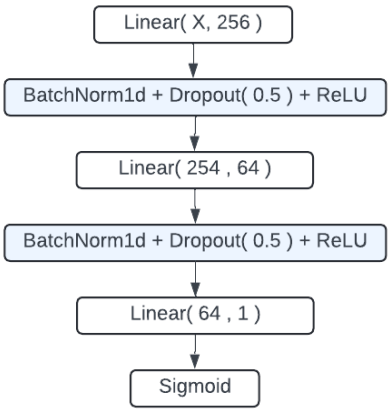


Figure B.3: Head of the Neural Network classification model for percentage Error and Standard Points

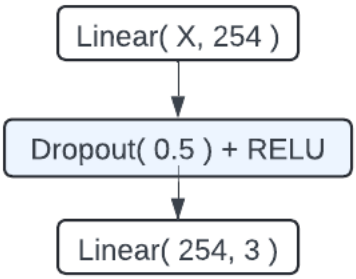


Figure B.4: Head of the Neural Network classification model - interior and exterior

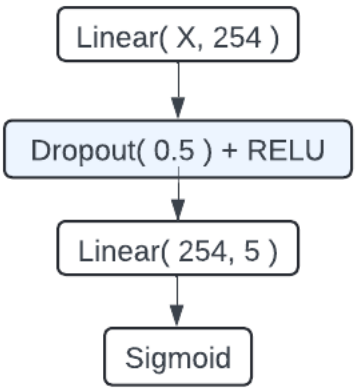


Figure B.5: Head of the Neural Network Classification Model - room type