Cities 133 (2023) 104107

Contents lists available at ScienceDirect

Cities

journal homepage: www.elsevier.com/locate/cities

Using social media images for building function classification

Eike Jens Hoffmann [a], Karam Abdulahhad [b], Xiao Xiang Zhu [a,*]

[a] Technical University of Munich, Chair of Data Science in Earth Observation, Arcisstraße 21, Munich 80333, Germany
[b] German Aerospace Center, Remote Sensing Technology Institute, Muenchener Straße 20, Weßling 82234, Germany

ARTICLE INFO

Keywords:
Social media image analysis
Big data analytics
Building function classification
Urban land use

ABSTRACT

Urban land use on a building instance level is crucial geo-information for many applications yet challenging to obtain. Steet-level images are highly suited to predict building functions as the building façades provide clear hints. Social media image platforms contain billions of images, including but not limited to street perspectives. This study proposes a filtering pipeline to yield high-quality, ground-level imagery from large-scale social media image datasets to cope with this issue. The pipeline ensures all resulting images have complete and valid geotags with a compass direction to relate image content and spatial objects.

We analyze our method on a culturally diverse social media dataset from Flickr with more than 28 million images from 42 cities worldwide. The obtained dataset is then evaluated in the context of a building function classification task with three classes: Commercial, residential, and other. Fine-tuned state-of-the-art architectures yield F1 scores of up to 0.51 on the filtered images. Our analysis shows that the quality of the labels from OpenStreetMap limits the performance. Human-validated labels increase the F1 score by 0.2. Therefore, we consider these labels weak and publish the resulting images from our pipeline and the depicted buildings as a weakly labeled dataset.

1. Introduction

While urban planning used to be performed as a top-down approach based on a master plan with zoning, new processes are needed to cope with rapid urban development in the global South (Watson, 2009). Historically, urban planning practices were developed in the global North with assumptions and aims that do not hold true nowadays or are outdated, like separations of income groups and accessibility for individual transport. Moreover, land use data is crucial for the evaluation of existing zoning provisions (American Society of Planning Officials, May 1950). For example, to know the demand for public transport, urban planning requires accurate numbers of citizens living in an area. These numbers can be estimated from land use data in combination with building heights. On the most fine-grained level, this data is calculated for each building individually and presumes information about the building's function. However, due to the rapid development, monitoring the *status quo* of building functions becomes infeasible. Automatic, data-driven methods can help to fill this gap. Building function classification is the task of automatically identifying the settlement type of a given building, e.g., is it a residential or commercial building? Traditionally, this process is performed manually, which is highly resource-consuming, and it cannot catch up with the size and speed of development of modern cities. To cope with this issue, automatic methods are applied, where they mainly consume air-view images, such as aerial or satellite images (Huang et al., 2018a; Zhang et al., 2019). Although this kind of data is of high quality, it has inherent ambiguities from a nadir view looking at rooftops.

During the last two decades, we have seen a tremendous increase in social media usage: Its data is ubiquitous, cheap, and easy to collect. It has become an essential and valuable source of information for many applications and scenarios (Kruspe et al., 2021). For example, it can serve as a proxy if authoritative data is missing or help to discover new phenomena, particularly in locations and populations where data from traditional sources are lacking (Lopez et al., 2019). In our task, social media data shows promising features to augment traditional air-view data sources. First, it offers a ground-level view, which means a finer-grained and different perspective data source. Second, it is a more up-to-date source of information or even a real-time source of information. Third, it is a huge source of cheap data. The only restriction in our scenario is that we need geotagged social media data. Fortunately, this is the case for a considerable share of data coming from social media channels such as Twitter or Flickr. For example, around 1 % of all tweets

* Corresponding author.
E-mail address: xiaoxiang.zhu@tum.de (X.X. Zhu).

https://doi.org/10.1016/j.cities.2022.104107
Received 27 April 2022; Received in revised form 25 September 2022; Accepted 10 November 2022
Available online 10 December 2022

are geotagged (Sloan et al., 2013), i.e., given that around 500 M[1] tweets are published per day, 5 M among them are geotagged. Flickr does not disclose its photo statistics in detail but announces having *billions of photos already online.*[2] By aligning geotagged social media content with open Volunteered Geographical Information (VGI) systems, such as OpenStreetMap (OSM), we could decode social media posts (e.g., tweets, images, etc.) to specific places on Earth and, hopefully, to certain buildings. However, one should not take social media as a no-cost source of information. One should be careful when dealing with social media as a primary source of information, where it is a noisy and uncontrolled data source. In addition, it is a sparse source, where social media equally cover not all spots on Earth. For example, Flickr photos are mainly coming from city centers and hotspots.

1.1. Related work

Generally, urban land use classification is a challenging task: no matter at which spatial level it is performed, inherent ambiguities exist. At the most fine-grained level, at the building instance level, it is often hard to decide which function a building serves. The task of building function classification has been approached with different data sources: The most intuitive one is street-level imagery showing the building façades. Alternatively, remote sensing data, especially optical imagery, is suited to predict building functions based on roof appearance and spatial context. Another potential data source is geotagged social media text messages, which can be analyzed with natural language processing or with pattern detection in metadata. Last but not least, taxi trajectories, mobile network usage, and point-of-interest databases have been used for building function classification. The following paragraphs present selected publications concerning the different data modalities.

Several studies investigated the feasibility of building façade images to address this problem. There are two primary sources for such ground-level image data: first, commercial ground-level data like Google Street View or Mapillary, and second, social media platforms like Facebook, Instagram, or Flickr.

Especially Google Street View is a preferred source for this task as its data is accessible using an API enabling the user to define the position, heading, pitch, and field-of-view. Additionally, Google has its own standardized hardware to capture street view images and a tailored image processing pipeline to generate high-quality imagery on a large scale. In combination with Google Places data, Google Street View data allows fine-grained store classification (Movshovitz-Attias et al., 2015). This work builds upon Google Map Maker ontology and a GoogLeNet architecture trained on a global sample of Google Street view. Access to Google Places is limited for research outside of Google, and Google Places focuses on points of interest (POIs) and does not include data about residential buildings. Alternatively, building footprints from OpenStreetMap (OSM) can also have semantic data, including details about building functions. This information can be used to label buildings shown in Google Street View images and hence, provides an additional way to predict land use on a building instance-level (Kang et al., Nov. 2018). The comprehensive coverage of buildings by Google Street View allows multiple images from different perspectives for a single building. This data richness can be used in a multi-modal architecture to include information from different sides while obtaining the labels from OSM (Srivastava et al., 2020). Beyond land use classification information encoded in Google Street View images can be used to infer socioeconomic characteristics (Gebru et al., 2017) or to map urban green in terms of tree detection and positioning (Laumer et al., 2020).

However, the terms of service of Google Street View prohibit scraping, downloading, or storing images obtained using the API. This legal constraint limits the applicability of Google Street View data in research projects and requires analyzing other data sources, e.g., social media image platforms. While Facebook and Instagram do not open their data for such purposes, Flickr turned out to be a valuable image source. They provide an easily accessible API and encourage their users to share photos with creative commons license. While early works on land use classification with Flickr images used bag-of-visual-word features for classification (Leung & Newsam, 2012), more recent studies benefited from advancements in computer vision with CNNs and proposed land use classification using a scene and an object detection stream in parallel (Zhu et al., 2019). On a larger spatial scale, Flickr has been used for mapping and understanding landscape aesthetics, either manually (Langemeyer et al., 2018) or based on CNNs (Havinga et al., 2021; Salem et al., 2020). Another field of application is flood-level estimation. By formulating this problem as an object detection task with Mask R-CNN it has been shown that these images help to predict discrete levels of flooding (Chaudhary et al., 2019). If social media images are used for a specific application, dealing with massive variations in motifs and scenes is crucial.

Other data sources with a dedicated purpose but limited spatial extent can be a better option in some cases. For example, images from Geograph project[3] are captured in a systematic way to cover Great Britain and Ireland. It aims to have at least one representative image for every square kilometer on both islands. These images can be used for predicting urban land use in London with object bank features (Fang et al., 2018; Li et al., 2010). Apart from Flickr, Twitter is also a social media data source providing geo-located information with textual features. Although Twitter restricted its geotagging feature in June 2019, it is still a valuable source of geospatial data (Kruspe et al., 2021). To predict building functions, it can be sufficient to have a set of geotagged tweets and build a classifier using their metadata (Huang et al., 2018b). As tweets contain mainly text, the inherent linguistic features have also shown potential to help in urban land use classification on a building instance-level (Häberle et al., 2019) as well as on a venue level (Terroso-Saenz & Munoz, 2020). Furthermore, geo-located Twitter data reveal patterns in language use and provide insights into socioeconomic factors when related to demographics (Bokányi et al., 2016). When used in combination with Flickr data, a correlation between socioeconomic factors and park visits shows up (Hamstead et al., 2018).

1.2. Contribution

In this paper, we tackle the problem of building function classification using social media images. To our best knowledge, this is the first method that relates the image content to individual buildings. It works on real-world, large-scale image datasets by establishing a rigid filtering pipeline that eliminates noisy, irrelevant, and non-geotagged photos. In contrast to other works, it is fully automatic and requires no manual selection or interaction (Chen et al., 2021b). After that, a Convolution Neural Network (CNN) is fine-tuned for a multi-class classification downstream task. This study mainly considers three classes of buildings from OSM, namely residential, commercial, and other. The main contribution of this paper can be summarized in the following points:

- Building function classification using weakly labeled Flickr images.
- A content-based automatic filtering pipeline to eliminate irrelevant and noisy Flickr photos from large-scale and real-world datasets
- A human-validated subset of Flickr photos for testing.

2. Methodology

Our method uses social media images to classify building settlement types. We follow a content-based approach, which can identify the main visual patterns for each class.

[1] https://www.internetlivestats.com/twitter-statistics/.

[2] https://www.flickr.com/jobs.

[3] https://www.geograph.org.uk/.

2.1. Social media image filtering pipeline

Social media images cover different content and motifs, including but not limited to photography, digital art, and cartoons. However, given a task like building function classification, most images do not help solve the task. For our task, an image must have three features:

1. Shows a building
2. Has a valid geotag
3. Has a known compass orientation

A filtering pipeline needs to identify all images fulfilling these three criteria in a social media image dataset. Additionally, it must account for big data to work on datasets with millions of images.

Fig. 1 shows the pipeline used in this study. It consists of five steps, starting with Google Street View similarity filtering and object detection filtering. These two steps together ensure that the first criterion is matched. We validate geotags in the next two steps: first, with a heuristic that discards images whose location is not unique. If another image is at precisely the same position, likely, that the geotag was manually edited. Second, we download the metadata for each remaining image and check if it contains a compass orientation. This step serves as a stricter check for the second criterion and ensures the last criterion. Finally, we use the geotag for spatial referencing with OSM buildings, including the compass direction.

2.1.1. Google street view similarity filtering

This first step is a coarse filtering step aiming at finding images that are potentially helpful for building function classification. Previous studies showed the relevance of façade images to predict building functions (Kang et al., Nov. 2018; Srivastava et al., 2020). Therefore, this step is formulated as an image retrieval problem with a sample of Google Street View images as seed dataset S and a social media dataset D.

Features from deep neural networks are well suited for finding structurally similar images. As they aggregate information with every layer, the final layers of a network are an abstract representation of the whole image. For example, the deep features of VGG16 (Simonyan & Zisserman, Apr. 2015) have been successfully applied in different domains for image retrieval (Ge et al., Jul. 2018; Ha et al., Aug. 2018; Liu et al., 2019; Wang et al., Oct. 2018).

In this study, features are taken from the last hidden layer of a VGG16 network trained on ImageNet (Russakovsky et al., Dec. 2015). This process yields feature vectors $v \in \mathbb{R}^{4096}$. To assess similarity between pairs of images i_1, i_2, the cosine similarity s_{cos} is calculated based on the feature vectors v_1, v_2:

$$s_{cos}(v_1, v_2) = \frac{v_1 v_2^T}{\|v_1\| \|v_2\|} \tag{1}$$

For efficient calculation, the features for all images of the seed dataset are calculated beforehand. Then, the features for all social media images are computed batch-wise, and we calculate the pair-wise cosine similarity between the batch and the seed dataset. For each social media image in the batch, we save the maximum similarity against all seed images, called the similarity parameter p_{sim}:

$$p_{sim}(v_s) = max(\{s_{cos}(v_1, v_s), \ldots, s_{cos}(v_n, v_s)\}) \tag{2}$$

A threshold t_{sim} is set as a minimum similarity value and all social media images with $p_{sim} < t_{sim}$ are discarded.

2.1.2. Object detection filtering

The previous step is a fast check for structural similarity to a given seed dataset but does not ensure that the social media images contain a building façade. Therefore, this step uses an object detection algorithm to find all objects in the images that passed the previous filter.

Applying the object detection algorithm yields a list of objects for each image. If this list contains either a *house* or a *building* it is a candidate for passing this filter. Each detected object comes with a size relative to the image and a confidence score. Based on these variables, there are two thresholds for adjusting if a candidate image passes the

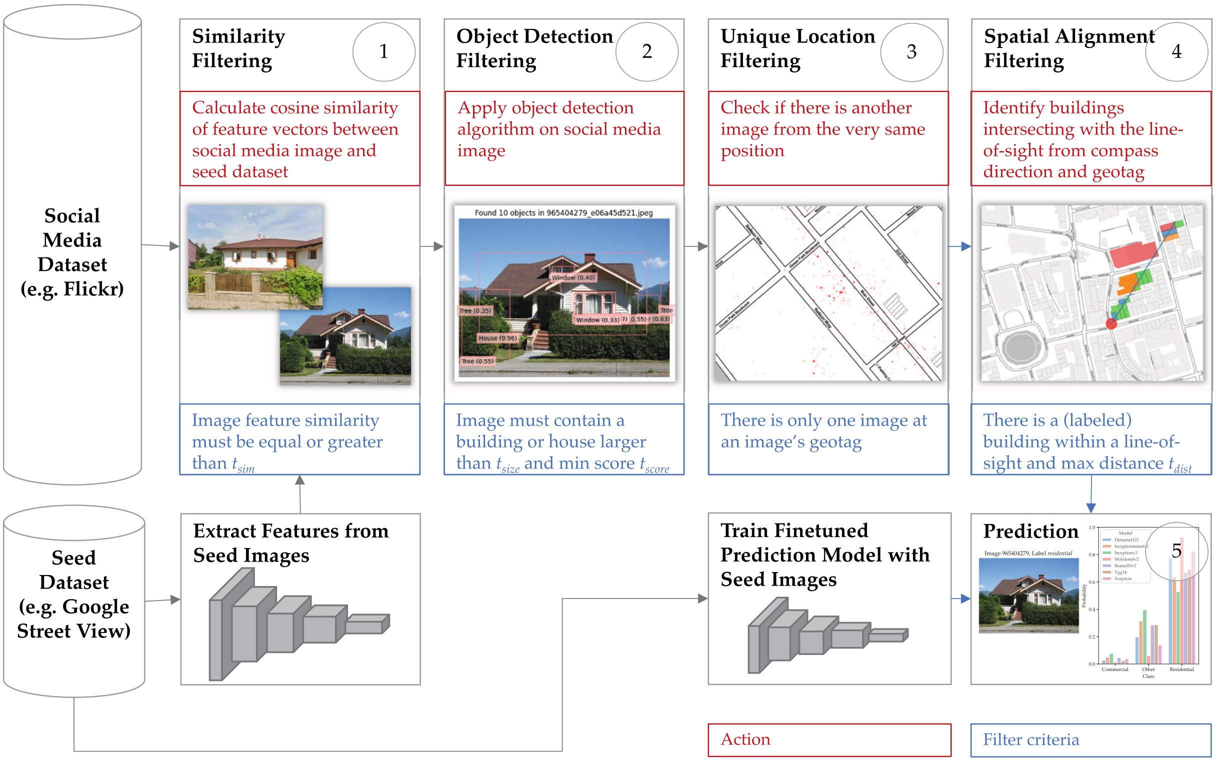

Social Media Dataset (e.g. Flickr)
Similarity Filtering
1
Calculate cosine similarity of feature vectors between social media image and seed dataset
Image feature similarity must be equal or greater than t_{sim}
Object Detection Filtering
2
Apply object detection algorithm on social media image
Found 10 objects in 965404279_e06a45d521.jpeg
Window (0.40)
Tree (0.35)
Window (0.33)
(0.55)
(0.63)
Tree
House (0.96)
Tree (0.55)
Image must contain a building or house larger than t_{size} and min score t_{score}
Unique Location Filtering
3
Check if there is another image from the very same position
There is only one image at an image's geotag
Spatial Alignment Filtering
4
Identify buildings intersecting with the line-of-sight from compass direction and geotag
There is a (labeled) building within a line-of-sight and max distance t_{dist}
Seed Dataset (e.g. Google Street View)
Extract Features from Seed Images
Train Finetuned Prediction Model with Seed Images
Prediction
5
Image 965404279, Label residential
Model
Densenet121
Inceptionresnetv2
Inceptionv3
Mobilenetv2
Resnet50v2
Vgg16
Xception
Probability
0.0
0.2
0.4
0.6
0.8
1.0
Commercial
Other
Residential
Class
Action
Filter criteria

Fig. 1. Filter pipeline for extracting Street View-like images from Flickr image database.

filter: t_{size} and t_{score}. Only if there is a building or a house with a size parameter p_{size} that is larger than t_{size} and has a confidence parameter p_{score} higher than t_{score} the image is passed to the next step.

2.1.3. *Unique location filtering*

The previous steps confirmed that the image content is relevant for the given task. Now, this step focuses on the geotag of the image. Geotags can be created in two different ways: either automatically by a GPS sensor of the camera or manually by the user.

This filter is a heuristic to identify images that have been manually tagged. If users have to pick locations of images by hand, they tend to do it batch-wise, tagging multiple images simultaneously. Otherwise, images tagged using a GPS sensor will have slight differences in the position even if the photographer has not moved. GPS sensors constantly update their location estimate based on how many GPS satellites are available. Therefore, having two images with precisely the same position is a strong indicator that their geotag has not been measured by a GPS sensor but manually added. In such cases, there is no compass orientation in the EXIF data, and hence, this image can be omitted for the subsequent step.

More formally, an image i from a set of images I with location $l(i)$ passes this filter if

$$\forall j \in I, j \neq i \nexists l(i) = l(j) \qquad (3)$$

A note on implementation: a sequential scan for each image is not feasible to make this step computationally efficient. If naïvely done, the geotag for each image needs to be compared with all geotags in the database. A geospatial index decreases the necessary checks by excluding geotags far away. Using an R-tree (Guttman, 1984) allows us to find the images in a very close neighborhood, and a subsequent check on true equality is performed only on the geotags of these images.

2.1.4. *Image direction filtering*

This step is based on metadata of images, so-called EXIF data. EXIF is a standard established by the Camera and Imaging Products Association (CIPA) and the Japan Electronics and Information Technology Industries Association (JEITA) (Camera &amp & Imaging Products Association, May 2019). It defines fields for saving details about images, including the date and time of capturing, camera model, and camera settings. Moreover, it specifies how data from GPS sensors can be incorporated. This data can be a position of longitude and latitude and a compass direction.

For our pipeline, we assume that the social media database does not contain the original images, including the EXIF metadata, but only a downsampled variant without original EXIF data. Therefore, we download the EXIF data for all images passing the previous filters as an intermediate step. Once all EXIF data are available, this step checks if the tag *GPSImgDirection* is present and rejects all images that do not have this tag.

Knowing the position where an image was taken is a necessary precondition, but only with the compass direction, a geospatial reference becomes feasible. Both information allows for calculating a line of sight, which is crucial for the next step.

2.1.5. *OSM reference building filtering*

This final step establishes a connection between buildings shown in an image and their representations in OpenStreetMap (OSM). OSM is a Volunteered Geographic Information (VGI) platform meaning that users contribute mapping data in a Wikipedia-like style. OSM provides guidelines on how this data should be structured and semantically enriched, but there is no strict enforcement. Therefore, tags for buildings are optional; just the building footprint coordinates are mandatory if added to OSM. OSM's guidelines specify three different tags that can be added for indicating a building function: *building*, *amenity*, and *shop*.

To summarize the information from all three tags, we use a mapping scheme that assigns each possible value according to OSM's guidelines of each tag to one of *commercial*, *residential*, and *other*. If more than one of these tags, *building*, *amenity*, and *shop*, is present, we make sure that they do not disagree. In case of disagreement, the building is not mapped to any class. If only one tag or all available ones agree on the same mapped class, then this building gets this class.

We use position and compass orientation to create a line of sight. All buildings' polygons intersecting the line of sight are possible candidates for the building shown in an image. We select the building with the closest distance to the position as the reference building in the picture and set this as parameter p_{dist}. Based on this parameter, we add a fourth threshold t_{dist} to analyze the effect of the distance.

For evaluation, we add another filtering step that discards all images assigned to a building without a semantic label.

2.1.6. *Filtering pipeline summary*

Having the pipeline in this order enables a content-first strategy while keeping the computational effort low. Additionally, the number of hyperparameters is small with four thresholds:

1. minimum seed similarity t_{sim}
2. minimum object size t_{size}
3. minimum object score t_{score}
4. maximum building distance t_{dist}

2.2. *Fine-tuning CNN architectures for building function classification*

To classify buildings shown in the social media images we fine-tune six state-of-the-art CNN architectures (DenseNet (Huang et al., Jan. 2018), InceptionV3, (Szegedy et al., Dec. 2015) MobileNetV2, (Sandler et al., Mar. 2019) ResNetV2, (He et al., Jul. 2016) VGG16, (Simonyan & Zisserman, Apr. 2015) Xception, (Chollet, Apr. 2017)). Starting with weights from ImageNet (Russakovsky et al., Dec. 2015) we applied a two-step approach to adapt the models for building function classification (Hoffmann et al., Jan. 2019). We start with ImageNet models without the classification head and add a dense layer with three outputs to predict each of the aforementioned homogenized OSM mapping scheme: *commercial*, *other*, and *residential*. Please note that we fine-tune the models on the Google Street View seed dataset and use social media images only for inference to predict building functions.

As a first step, all layers are frozen, and only the new, randomly initialized layer is trained with a learning rate of $lr = 10^{-4}$ for at most 16 epochs. Hence, the new layer is adapted to the current weights, and there is no risk of collapsing weights when trained on the entire network. A checkpointing mechanism makes sure that after training, the model with the lowest validation loss is restored and used for the next step. This method prevents overfitting in both steps.

After convergence of the newly added layer, the whole model is set as trainable and fine-tuned in a second step with a learning rate of $lr = 10^{-5}$. Again, we apply the checkpoint mechanism and create the final model based on the one with the lowest validation loss during training of 16 epochs.

2.3. *Human label validation*

To validate the labels obtained from OpenStreetMap buildings, we asked a group of humans to verify the labels given to an image. Given a question if an image contains a *commercial/other/residential* building, they had to choose between three options: *yes*, *unsure*, or *no*. If *no* was selected, users were asked for the correct label in their opinion.

As our classification scheme does not include mixed-use labels, we asked our voters to opt for *unsure* if no clear label could be assigned. Our system requested three votes from different humans for each image to make the votes more reliable. Once an image received three votes, it was not shown to any other user again. The users were not restricted in the number of images to vote on.

3. Experiments

We first introduce the two datasets used in this study and describe the results of the different filtering steps. Moreover, we show the results of a Google Street View trained model on filtered Flickr images and dive deeper into the prediction performance by including results from the human validation setup.

3.1. Datasets

We evaluate our method using two datasets: First, a sample of Google Street View (GSV) images featuring buildings with known functions, and second, a Flickr image dataset captured in 42 cities with global distribution.

3.1.1. Google street view dataset

The Google Street View dataset consists of 43,392 building facade images, distributed to 14,512 commercial, 14,184 other, and 14,696 residential buildings. We apply a faster R-CNN (Ren et al., Jan. 2016) trained on OID v4 (Kuznetsova et al., Jul. 2020) to detect objects on all images and discard all images that do not show a *building* or *house*. This combination of architecture and dataset has the best trade-off between accuracy and speed (Huang et al., Apr. 2017). This yields a refined dataset of 7698 images (2743 commercial, 2333 other, and 2622 residential).

This Google Street View dataset is used in two ways: first, as a seed dataset for finding structurally similar images in the social media dataset, and second, for fine-tuning state-of-the-art CNN architectures on the given task.

3.1.2. Flickr social media dataset

We collected Flickr image data in 42 cities across the globe to cover different cultures and continents. The images were obtained by querying the Flickr API with small random bounding boxes inside these regions of interest. With this approach, we harvested 28,818,438 images.

Table 1 shows the number of images per city. The number of images per city correlates with the user distribution of Flickr, so we see the highest number of images in London (~4.0 M images). Second, New York City has ~2.3 M images, and third, Los Angeles with ~1.9 M images. Except for Dongying, we found more than 5000 images in every city. There is evidence that Dongying is a ghost city, meaning that the housing capacity outnumbers the number of inhabitants by far (Leichtle et al., Nov. 2019).

Table 1
Number of Flickr images per city.

City	#Images
Amsterdam	1,147,657
Beijing	358,393
Berlin	929,508
Cairo	110,297
Cape Town	165,848
Changsha	8051
Cologne	610,185
Dongying	153
Guang Zhou	81,585
Hong Kong	964,733
Islamabad	9779
Istanbul	259,141
Jakarta	204,792
Kyoto	668,547
Lisbon	463,992
London	3,978,803
Los Angeles	1,979,163
Madrid	709,029
Melbourne	661,921
Milan	735,996
Moscow	569,651
Mumbai	140,495
Munich	391,798
Nairobi	32,262
Nanjing	24,411
New York City	2,351,955
Paris	1,344,000
Qingdao	11,960
Rio De Janeiro	425,874
Rome	570,033
San Francisco	1,744,662
Santiago	269,656
Sao Paulo	729,197
Shanghai	330,229
Shenzhen	51,893
Sydney	730,823
Tehran	21,999
Tokyo	1,361,486
Vancouver	834,973
Washington D.C.	1,139,602
Wuhan	25,754
Zurich	288,903

3.2. Filtering pipeline results

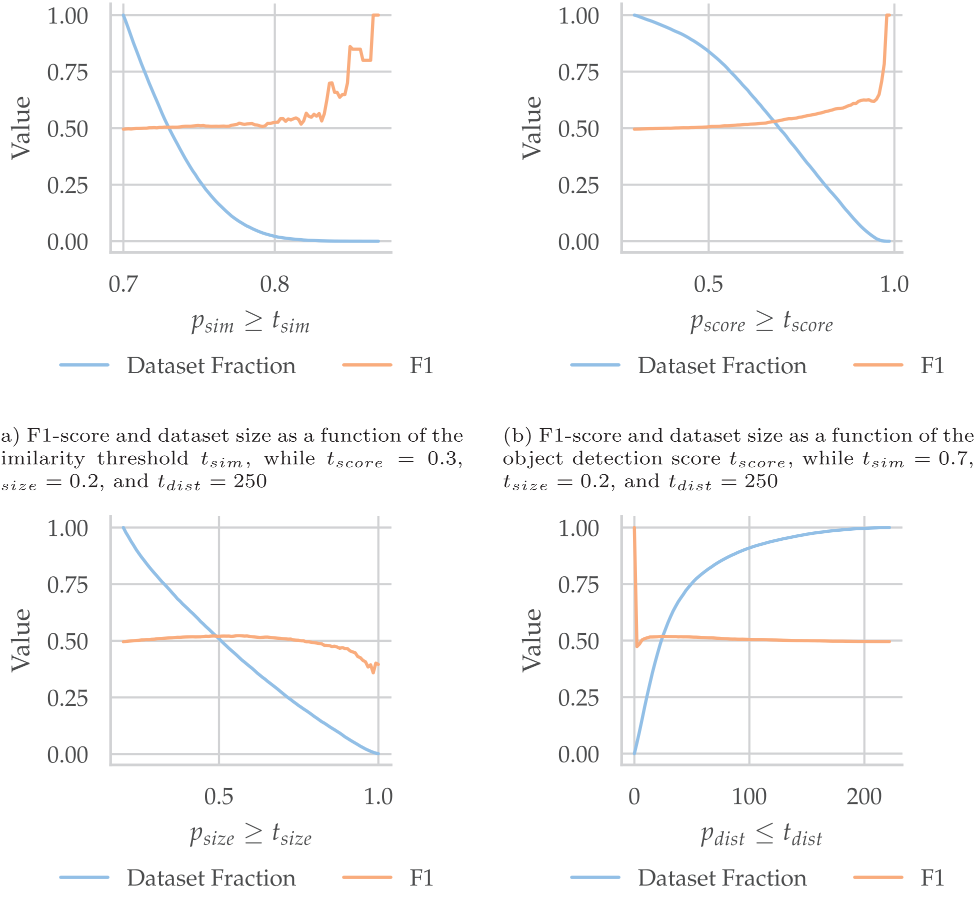
We evaluate our pipeline end-to-end by analyzing the effects of the four hyperparameters on the F1 score. For an architecture-independent evaluation, we calculated the mean probability vectors of all six models for each image. Using mean probability vectors of six models eliminates artifacts from single models and allows more general conclusions. Fig. 2 shows the F1-scores and the remaining dataset size as functions of a threshold. Computing the F1 score requires working on the final output of the pipeline, with each image being assigned to an individual building. Hence, the complete dataset of 100 % is based on 26,381 images, 8070 labeled as commercial, 9171 labeled as other, and 9140 labeled as residential.

Our analysis is ordered by the appearance of the hyperparameters in the pipeline. First, there is the similarity threshold t_{sim} setting how similar a social media image must be compared to the seed dataset (Fig. 2a). Between 0.70 and 0.80 there is little difference in the resulting F1-score: It is almost constant between 0.50 and 0.52. At the same time, the dataset size decreases from 100 % to 2 %. The F1 scores show the first peak at $t_{sim} = 0.83$ with an F1 score of 0.70 and a corresponding dataset size of 0.08 % (23 images in total). For thresholds higher than 0.85 F1 scores become unreliable as the number of images is seven or less. Figs. 3, 4, and 5 show examples of Flickr images having a $p_{sim} = 0.50$, $p_{sim} = 0.75$, and $p_{sim} = 0.90$.

Fig. 2b shows how the F1 score is affected by the object detection score p_{score}. The figure starts with $t_{score} = 0.30$ because objects with lower scores are not reported by the implementation we used. It increases slightly starting from $t_{score} > 0.30$ with an F1 score of 0.50 up to 0.63 at $t_{score} = 0.93$. At the same time, the dataset decreases from 100 % to 3.6 % (this is equal to 930 images). Setting $t_{score} > 0.965$ yields an increase in F1 score to 0.70, but with only 0.2 % or 56 images being considered.

The second parameter from the object detection filtering is t_{size}, the minimum size of the *building* or *house* to be found in an image (Fig. 2c). Using a threshold $t_{size} = 0.2$ yields a F1 score of 0.50. Increasing the threshold up to $t_{size} = 0.56$ results in a higher F1 score of 0.52, which is the highest possible value. Raising the threshold further decreases the F1 score down to 0.40 at $t_{size} = 1.0$. At the peak of $t_{size} = 0.57$ the remaining dataset consists of 11,234 images (44 %).

As a last parameter in the pipeline, there is the distance between a photographer's position and the next building in the compass direction p_{dist}. Fig. 2d depicts the F1 score as a function of the distance. Please note that the threshold is an upper limit. Setting $t_{dist} = 0.0$ yields a F1 score of 1.0 based on a single image. Increasing to $t_{dist} = 2.2$ provides a first realistic value of 0.48 calculated on 4.8 % of the dataset. Raising the threshold further to $t_{dist} = 40.32$ results in the highest possible F1 score of 0.52. At this point, 13,999 images, 69 % of the dataset, are included. Higher thresholds lead to a slight decrease of the F1 score down to 0.50 at $t_{dist} = 222$.



(a) F1-score and dataset size as a function of the similarity threshold t_{sim}, while $t_{score} = 0.3$, $t_{size} = 0.2$, and $t_{dist} = 250$

(b) F1-score and dataset size as a function of the object detection score t_{score}, while $t_{sim} = 0.7$, $t_{size} = 0.2$, and $t_{dist} = 250$

(c) F1-score and dataset size as a function of the relative size of a detected building t_{size}, while $t_{sim} = 0.7$, $t_{score} = 0.3$, and $t_{dist} = 250$

(d) F1-score and dataset size as a function of the distance to the next labeled building t_{dist}, while $t_{sim} = 0.7$, $t_{score} = 0.3$, and $t_{size} = 0.2$

Fig. 2. Effect of filtering pipeline parameters on prediction results and remaining dataset size.

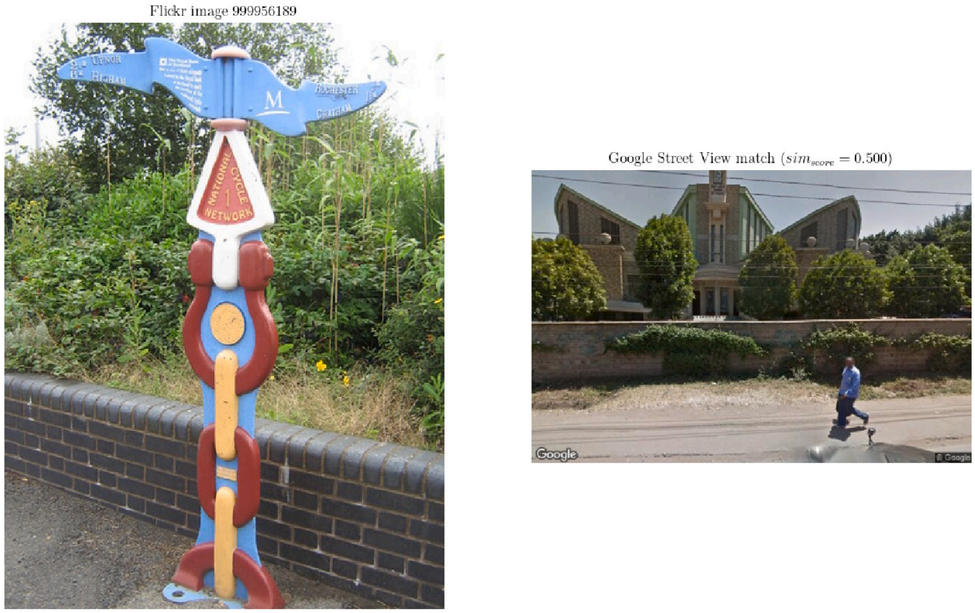


Fig. 3. Sample for Flickr image having a $p_{sim} = 0.50$.

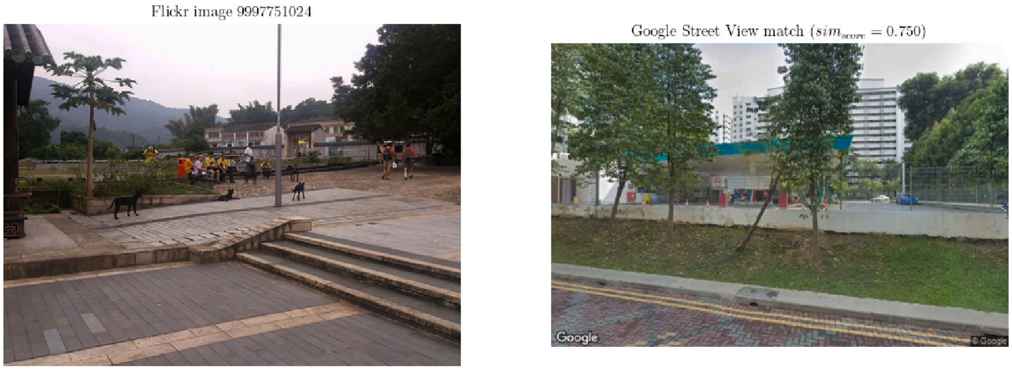


Fig. 4. Sample for Flickr image having a $p_{sim} = 0.75$.

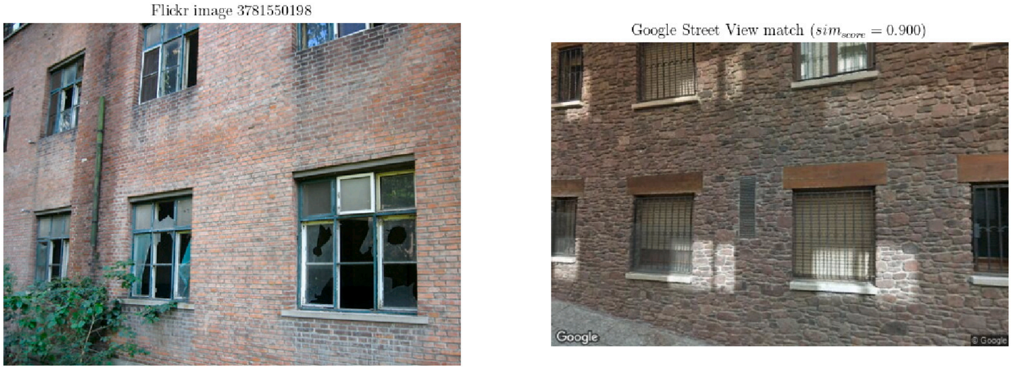


Fig. 5. Sample for Flickr image having a $p_{sim} = 0.90$.

Overall, the hyperparameters do not have too much influence on the prediction quality. We see that the F1 score is primarily stable at around 0.5. Just in the case of strict thresholds, there are some exceptions: e.g., setting $t_{score} = 0.965$ yields an F1 score of 0.70. Adjusting thresholds has more effects on the dataset size: On the one hand, fixing too strict thresholds yields a low number of images. On the other hand, this has a significant effect on the runtime of the whole pipeline. The more images a filter step at the beginning, the higher the overall computational time. This trade-off needs to be taken into account when applying the filtering steps.

Table 2 illustrates the number of images remaining after each filtering step when setting $t_{sim} = 0.70$, $t_{score} = 0.3$, $t_{size} = 0.2$ and $t_{dist} = 250$. Additionally, it shows how long it takes to process a single image in a filter step in our setup. While the exact times will change with different setups, the relative comparison allows an assessment of the effectiveness.

Similarity filtering reduces the remaining images to less than 6 % of the original dataset at high speed. Discarding all images that do not show a house or a building yields 891,861 images (3.09 % of all images in the original dataset). However, this second step of content filtering takes more than 25 times longer than the similarity check.

Table 2
Number of Images remaining after each Filtering Step when using $t_{sim} = 0.70$, $t_{size} = 0.2$, $t_{score} = 0.3$, and $t_{dist} = 250$. Execution time per image sample in seconds.

Filtering step	#Images	% of dataset	Execution time [s]
Flickr LCZ42 Dataset	28,818,438	100.00 %	
Similarity filtering	1,635,592	5.68 %	0.0236
Object detection filtering	891,861	3.09 %	0.6319
Unique location filtering	457,670	1.59 %	0.0002
Image direction filtering	88,593	0.31 %	1.3333
OSM building in line-of-sight	73,207	0.25 %	0.0008
Labeled OSM building in line-of-sight	26,381	0.09 %	

Ensuring that there is no other image from the same location filters out 743,731 images, which indicates that almost half of all images were manually tagged. Utilizing a spatial index makes this step the fastest of all filtering steps with 0.2 milliseconds, a hundred times faster than the similarity check. Out of the remaining 457,670 images, 88,593 have a compass orientation, which is 0.31 % of the original dataset. This step requires downloading additional data using the Flickr API; it takes 1.33 s. Please note that most of the time is spent waiting for the subsequent API request to prevent being blocked by the platform (1 s).

Checking if an OSM building footprint is within the line of sight kept 73,207 images, and limiting this to labeled OSM buildings gave 26,381 images. This result is 0.09 % of the whole dataset. This step again uses the spatial index, which results in the second-fastest check of all steps.

There can be more than one image per building, and several images cover especially touristic landmark buildings. The 6955 images from our filtering pipeline were mapped to 18,759 buildings. 5962 of them are *commercial*, 5138 are *other*, and 7659 are *residential*.

3.3. Prediction results

Table 3 summarizes the performance of all fine-tuned models on an image level. Class-wise they behave similar with higher recall values on *commercial* and *other* labeled images and a higher precision value for the *residential* class. One exception in this pattern is VGG16, which has the highest precision score for *other*. The mean F1 score for *commercial* is 0.51, which is slightly higher than the F1 score for *other*, 0.47, and *residential*, 0.37.

Residential buildings can appear as single-detached houses, townhouses, apartment blocks, or skyscrapers. While the first two forms of residential buildings are easy to predict, the latter can be easily confused

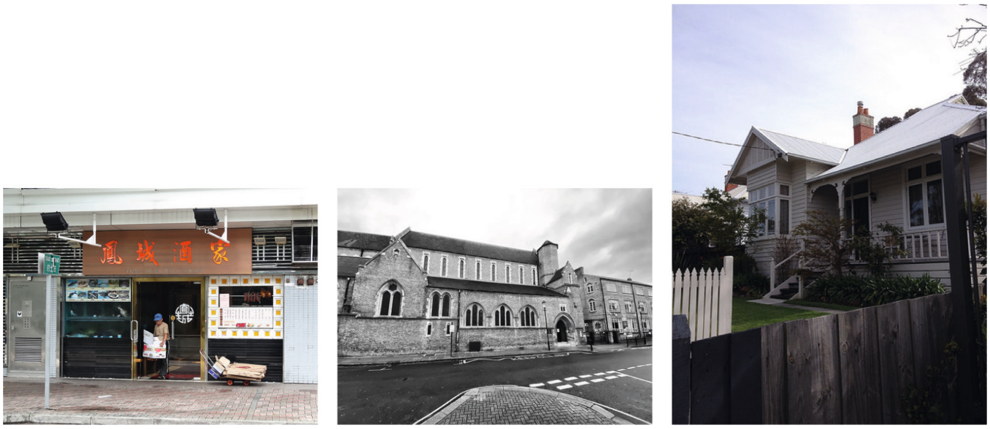
Table 3
Prediction results of fine-tuned Google Street View models on filtered Flickr images and on human validated subset. Class labels are abbreviated as highlighted in bold: ***Com**mercial*, ***Oth**er*, ***Res**idential*, and *Avg* stands for the weighted average based on the number of samples.

Architecture	Metric class	Filtered images			Human-validated images		
		F1	Prec	Rec	F1	Prec	Rec
DenseNet121	Com	0.52	0.43	0.66	0.76	0.65	0.91
	Oth	0.49	0.49	0.50	0.76	0.77	0.76
	Res	0.43	0.60	0.34	0.64	0.82	0.51
	Avg	0.47	0.50	0.481	0.72	0.75	0.73
Inceptionv3	Com	0.51	0.44	0.61	0.76	0.68	0.91
	Oth	0.51	0.45	0.58	0.74	0.67	0.76
	Res	0.39	0.63	0.28	0.55	0.87	0.51
	Avg	0.46	0.50	0.48	0.68	0.73	0.70
MobileNetv2	Com	0.49	0.45	0.54	0.76	0.72	0.81
	Oth	0.51	0.45	0.61	0.74	0.67	0.82
	Res	0.44	0.62	0.34	0.61	0.80	0.49
	Avg	0.46	0.50	0.48	0.71	0.73	0.71
ResNet50v2	Com	0.48	0.43	0.55	0.73	0.67	0.81
	Oth	0.45	0.44	0.46	0.69	0.70	0.68
	Res	0.45	0.53	0.40	0.58	0.65	0.53
	Avg	0.45	0.45	0.45	0.67	0.67	0.67
VGG16	Com	0.53	0.45	0.64	0.74	0.64	0.86
	Oth	0.35	0.58	0.25	0.61	0.87	0.47
	Res	0.55	0.52	0.58	0.68	0.62	0.74
	Avg	0.47	0.49	0.47	0.67	0.72	0.68
Xception	Com	0.53	0.42	0.69	0.75	0.64	0.91
	Oth	0.48	0.48	0.49	0.70	0.69	0.71
	Res	0.40	0.63	0.29	0.62	0.85	0.48
	Avg	0.46	0.50	0.47	0.69	0.73	0.70

with the other two classes. This hypothesis is one possible explanation for why we see a high precision for *residential* buildings but a lower recall.

All architectures show similar performance on the social media dataset. Concerning the weighted average, the Densenet121 and VGG models show the best F1 score of 0.47, but the worst model has an F1 score of 0.45 (Resnet50). Hence, the prediction errors are not model-specific but rather data issues. Figs. 6 and 7 illustrate positive and negative examples of building function prediction using nearby Flickr images. The façade of a restaurant near Hong Kong is correctly predicted as *commercial* (Fig. 6a), which matches the overall building function from OSM. Likely, the large Chinese characters are a highly distinctive feature. Fig. 6b is an interesting example showing that the prediction works on greyscale images as well. The church in the center of London is correctly predicted as *other*. The single-detached house in Fig. 6c is an ideal example of a *residential* building in the suburbs of Melbourne. However, the building is pictured from the side with a different perspective than Google Street View images, yet it is predicted as *residential*. All three photos show a perfect, real-world example for each class, i.e., a single building with a clear function in the center of the photo. Although the model was trained on Google Street View images, it can cope with different cultures, greyscale images, and changes in perspective. In the end, the model predicted the correct class for each building. In contrast to these examples, Fig. 7 shows three photos that are predicted with a different class than the actual label from OSM. The first example in Fig. 7a depicts an image centered in the sky and has some houses at the bottom. The image is mapped to a bar called *The Royal Oak* in London and hence, it is labeled as *commercial*. However, based on the image content, the DenseNet model predicts the buildings as *other*. A cross-check with Google Street View reveals that the compass direction is slightly off and the photo is mapped to the building, which is cut off at the right side. Hence, in this case, the image is mapped to the wrong building. Fig. 7b shows an image, which is mapped to a *other* building, but the model predicts it as a *commercial* one. This is a borderline case in our classification scheme: We define a train station as *other* because we see it as a part of public infrastructure. However, as most train systems are organized as companies, predicting it as *commercial* is also theoretically possible. Of course, the machine learning model is simply looking for image features to define class borders and in this case, the large sign a the top of the image is a strong hint toward a *commercial* usage. Finally, Fig. 7c depicts an image that is labeled as *residential*. Despite this label, the image shows the front of a famous restaurant in New York City. It is located on the ground floor of a multistory apartment house. Hence, the model's prediction is plausible from the image content but it misses the rest of the building and its predominant function. This is a limit of a single-label classification scheme, which cannot reflect mixed-used buildings. We investigate this data quality issue with the labels in the next section. It investigates the effect of OSM labels on classification performance based on human verification of the labels.

Fig. 8 shows an example of Flickr images and their corresponding buildings from OSM in the city center of Mumbai, India. Only eleven buildings are annotated with a function while the majority of buildings, 37, in red have no semantic annotation at all. This illustrates the potential of our method: Although social media images are an



(a) Flickr image showing a part of the *commercial* building Tower 2 (OSM ID 356349450) near Hong Kong. Photo *Untitled* by iombie is licensed under CC BY-NC-ND 2.0

(b) Flickr image showing the façade of *other* building Saint Mary's Church (OSM ID 264381580) in London. Photo ©PBWA Kensington and Chelsea by Ian Wood

(c) Flickr image showing a *residential* building in Spencer Road 23, Melbourne (OSM ID 875529280). Photo Sad house by Leonie Bourke is licensed under CC BY-NC-SA 2.0

Fig. 6. Examples of Flickr images with correct predictions for the building functions.

(a) Flickr image mapped to *commercial* building The Royal Oak in London (OSM ID 404678712). Photo ©London Sunset, Marylebone by Koji Moriya

(b) Flickr image mapped to an *other* building, a train station, near Kyoto (OSM ID 940194500). Photo ©IMG_7634 by vincent chang

(c) Flickr image showing *residential* OSM building The Packard in New York City (ID 541718706). Photo ©*Untitled* by Julie Roth

Fig. 7. Examples of Flickr images that yielded an incorrect prediction compared to the building label.

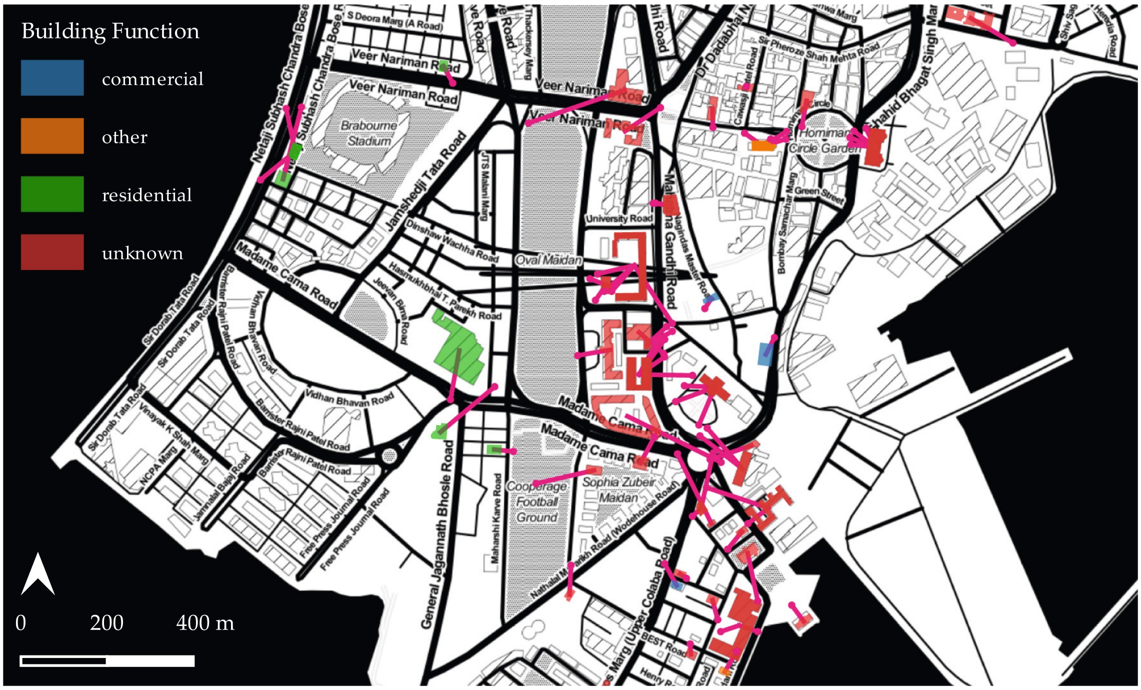


Fig. 8. Map of Flickr images in Mumbai spatially aligned with buildings. Magenta lines indicate the line of sight from an image toward a building. Buildings are colored based on their function as described in OpenStreetMap. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

opportunistic data source, they can still help to close gaps in VGI data sources. In this example, they could add additional geospatial knowledge to 70 % of the identified buildings.

3.4. Label verification results

For this experiment, we selected a random subset of 1500 social media images with OSM labels, 500 from each class, to be validated by humans. As we required three votes for each image, we understood how difficult the task is for a person seeing only the image and the label. Out of 1500 images, 756 images have total agreement on their label, and 744 received inconsistent votes.

Full agreement includes three *unsure* votes as well, so in Fig. 9 we focused on the images that received a clear vote, either *correct* or *wrong*. Overall, the accuracy of OSM is 69 %, but there are subtle differences between the three classes. *Commercial* has 63.5 % correct labels, which is the lowest value of all classes. On the other hand, *residential* images show 72.5 % correctness with *other* being similar (71.3 %).

To assess our models' true performance, we evaluate our models on the subset of images that received either complete agreement on the existing label or an entire agreement on a new label. The right part of Fig. 3 shows the F1 score, precision, and recall for this subset. The patterns concerning precision and recall described above are the same, but all values improved by 0.2.

Hence, our models yield good results if applied to clear data. In this case, the Densenet121 model yields the best F1 score of 0.72, with MobileNetV2 being second. The VGG16 model is among the worst, with an F1 score of 0.67, while it showed up in the first place on the filtered dataset. The Densenet121 model has the best generalized essential features for building function classification.

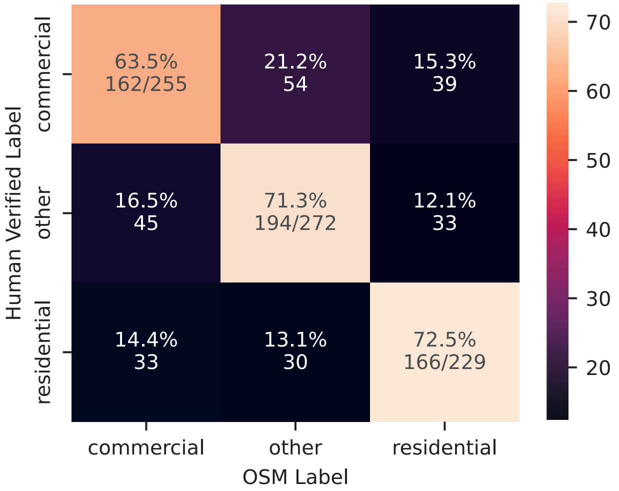


Fig. 9. Results from human validation of OSM labels as confusion matrix with only full agreement of human voters.

A big source of error is unclear, mislabeled, or mixed-used buildings. Considering that almost half of the images in the human-validated subset did not receive any consistent vote from humans, the performance of these models is sensible.

4. Discussion

4.1. Ambiguity of the task

Our simple classification schema of *commercial*, *other*, and *residential* works best if buildings have such a clear function. In historically grown city centers, mixed-use buildings are more common, with a retail store on the ground floor and apartments on the upper floors. Our three classes reach their limits and result in an error source in these cases. Especially if there is a large sign above the ground floor advertising a retail store, this will likely cause a misclassification. Additionally, the *other* class is not well defined. Serving as an alternative if none of *commercial* or *residential* truly fit, there are a lot of different patterns pointing to the same final decision. This fuzziness makes it hard for a CNN to predict this class.

4.2. Missed images in filtering pipeline

Although we sampled our Google Street View image dataset on a global scale, there are most likely types of buildings that are not covered. This lack will result in discarded images, although it shows a building and contains valuable geographic information. However, the prediction would likely be wrong in this case as the seed dataset for filtering, and the training dataset for fine-tuning are identical. Hence, there would be no benefit in including this image. Possible mitigation could be a more sophisticated sampling algorithm that includes rare building types.

The same applies to the object detection algorithm. If there are building types that were not in the training dataset of OID, images can be filtered out despite having a building inside, which would be correctly predicted.

4.3. Correctness of OpenStreetMap labels

OSM's primary goal is to provide an open geoinformation service for users to orientate, navigate, and find places of interest (POIs) for their needs. Therefore, *commercial* and *other* buildings providing any service for society are more likely mapped than *residential* buildings that have no general purpose. *Residential* buildings are often bulk mapped so that certain neighborhoods show a high level of completeness, while others do not have any building footprint at all.

Additionally, building functions may change: what used to be a church becomes an apartment building or is abandoned. The validity of labels depends on the activity of OSM's contributors. Hence, in regions with many active contributors, labels will be more up-to-date than in regions with very few contributors.

Last but not least, in areas with few active local contributors like Africa, OSM buildings are mostly mapped by remote users looking at aerial imagery and drawing building polygons accordingly. There will most likely be no semantic labels at all in such cases.

4.4. Completeness of OpenStreetMap building footprints

As a VGI platform, the completeness of OpenStreetMap buildings polygons varies a lot. If a building footprint in OSM is missing, our algorithm may assign an image to a building that is actually behind the one it shows.

4.5. Reference building calculation

Several images show street view perspectives, including more than one building. In such cases, our line-of-sight algorithm will check which building is the building of the image. Buildings on the left and the right will be ignored.

5. Conclusion and outlook

This study proposes a content-first filtering pipeline for social media image datasets to extract geospatial information on building functions. By applying five filtering steps, we can find relevant images with valid metadata for the given task and relate them to buildings within the line of sight. The order of the filter steps ensures scalability on large image databases. Moreover, our pipeline has only four hyperparameters for balancing runtime, and the number of images yielded without strong influence on final prediction results. Based on human validation of our image labels from OSM, we show that the limiting performance factor is rather the data quality of OSM labels than the models used for predictions. The resulting image dataset with corresponding OSM building IDs and labels is published as a benchmark dataset for urban land use using social media images and weak OSM labels. Additionally, we provide the human-validated subset with high-quality labels based on three independent votes.

In urban planning, this pipeline could be used for automatic checks if planned and actual building functions differ. Therefore, it can be used as a tool for gap analysis and land use optimization (Zhang & Huang, 2015). Moreover, building function classification is only one application among others. The resulting images from this pipeline enable studies on the attractiveness of buildings by the number of images. Analyzing the image content allows gathering insights into the parts and perspectives that are interesting to visitors. Hence, this poses a backward communication channel for a concept known as urban planning as communication (Innes & Booher, 2015). Further potential applications are the validation of existing VGI data, e.g., map data in OSM (Hoffmann et al., 2020) or labeling remote sensing data (Chi et al., 2017). Moreover, social media images go beyond the scope of Google Street View: while the latter is bound to streets, social media images can be taken from any position. Hence, there are no occlusions from trees, but rather it can retrieve images from pedestrian zones or other areas that are inaccessible to cars. These images can be used for improved LOD 2 and 3 building models with high-resolution textures and architectural details on balconies, windows, or doors. Such high-quality models are the basis for digital twins of cities, which will enhance the realization, operability, and management of cities (Shahat et al., 2021). Last but not least, with a different seed dataset, the pipeline allows for relating arbitrary images to objects on maps. Such objects could be trees for mapping urban green or infrastructures like street lights or fire hydrants.

Our pipeline still has many opportunities for refinement. While the cosine similarity measure against a seed dataset ensures fast processing speed, this image retrieval task can be enhanced with more sophisticated algorithms taking different aspects of an image into account (Chen et al., 2021a). One of the most rigid filtering steps is discarding all images without a compass orientation. Recent approaches that estimate the compass orientation based on aerial imagery could be of help to close this gap (Regmi & Shah, 2019; Shi et al., 2020; Vo & Hays, 2016). As the last step, we relate the image to a building using a line-of-sight. Fortunately, the EXIF metadata contains data about the focal length opening a possibility to calculate all buildings within the field of view. An image could be separated into patches with different buildings found during the object detection step. This step could yield predictions for many buildings from one image. Moreover, our classification scheme with *commercial*, *residential*, and *other* focuses on the most crucial classes for population estimation and disaster management. A more fine-grained, multi-level scheme could provide more insights into urban development, e.g., education, transportation, and health care. Another possible direction could be introducing multi labels to consider mixed-use buildings.

CRediT authorship contribution statement

Eike Jens Hoffmann: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Visualization. **Karam Abdulahhad:** Validation, Writing – original draft, Writing – review & editing, Supervision, Project administration. **Xiao Xiang Zhu:** Conceptualization, Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare no conflict of interest.

Data availability

The authors do not have permission to share data.

Acknowledgement

The work is jointly supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. [ERC-2016-StG-714087], Acronym: *So2Sat*), by the Helmholtz Association through the Framework of the Helmholtz Excellent Professorship "Data Science in Earth Observation - Big Data Fusion for Urban Research"(grant number: W2-W3-100), by the German Federal Ministry of Education and Research (BMBF) in the framework of the international future AI lab "AI4EO – Artificial Intelligence for Earth Observation: Reasoning, Uncertainties, Ethics and Beyond" (grant number: 01DD20001) and by German Federal Ministry for Economic Affairs and Climate Action in the framework of the "national center of excellence ML4Earth" (grant number: 50EE2201C).

References

American Society of Planning Officials. (May 1950). *Urban land use. 14*. CHICAGO.

Bokányi, E., Kondor, D., Dobos, L., Sebök, T., Stéger, J., Csabai, I., & Vattay, G. (2016). In *, 2(1). Race, religion and the city: Twitter word frequency patterns reveal dominant demographic dimensions in the United States* (pp. 1–9). Palgrave Communications.

Camera &amp, Imaging Products Association. (May 2019). *Exchangeable image file format for digital still cameras: Exif Version 2.32*.

Chaudhary, P., D'Aronco, S., Moy de Vitry, M., Leitão, J. P., & Wegner, J. D. (2019). Flood-water level estimation from social media images. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 4*(2/W5), 5–12.

Chen, W., Liu, Y., Wang, W., Bakker, E., Georgiou, T., Fieguth, P., Liu, L., & Lew, M. S. (2021). *Deep image retrieval: A survey*. arXiv preprint arXiv:2101.11282, 2021a.

Chen, Y., Sherren, K., Smit, M., & Lee, K. Y. (2021). Using social media images as data in social science research. *New Media & Society*, Article 14614448211038761.

Chi, M., Sun, Z., Qin, Y., Shen, J., & Benediktsson, J. A. (2017). A novel methodology to label urban remote sensing images based on location-based social media photos. *Proceedings of the IEEE, 105*(10), 1926–1936.

Chollet, F. (Apr. 2017). *Xception: Deep learning with depthwise separable convolutions*. arXiv:1610.02357 [cs].

Fang, F., Yuan, X., Wang, L., Liu, Y., & Luo, Z. (2018). Urban land-use classification from photographs. *IEEE Geoscience and Remote Sensing Letters, 15*(12), 1927–1931.

Ge, Y., Jiang, S., Xu, Q., Jiang, C., & Ye, F. (Jul. 2018). Exploiting representations from pre-trained convolutional neural networks for high-resolution remote sensing image retrieval. *Multimedia Tools and Applications, 77*(13), 17489–17515.

Gebru, T., Krause, J., Wang, Y., Chen, D., Deng, J., Aiden, E. L., & Fei-Fei, L. (2017). Using deep learning and google street view to estimate the demographic makeup of neighborhoods across the united states. *Proceedings of the National Academy of Sciences, 114*(50), 13108–13113.

Guttman, A. (1984). R-trees: A dynamic index structure for spatial searching. In *Proceedings of the 1984 ACM SIGMOD international conference on Management of data* (pp. 47–57).

Ha, I., Kim, H., Park, S., & Kim, H. (Aug. 2018). Image retrieval using BIM and features from pretrained VGG network for indoor localization. *Building and Environment, 140*, 23–31.

Häberle, M., Werner, M., & Zhu, X. X. (2019). Building type classification from social media texts via geo-spatial textmining. In *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium* (pp. 10047–10050). IEEE.

Hamstead, Z. A., Fisher, D., Ilieva, R. T., Wood, S. A., McPhearson, T., & Kremer, P. (2018). Geolocated social media as a rapid indicator of park visitation and equitable park access. *Computers, Environment and Urban Systems, 72*, 38–50.

Havinga, I., Marcos, D., Bogaart, P. W., Hein, L., & Tuia, D. (2021). Social media and deep learning capture the aesthetic quality of the landscape. *Scientific Reports, 11*(1), 1–11.

He, K., Zhang, X., Ren, S., & Sun, J. (Jul. 2016). *Identity mappings in deep residual networks*. arXiv:1610.02357 [cs].

Hoffmann, E. J., Wang, Y., Werner, M., Kang, J., & Zhu, X. X. (Jan. 2019). Model fusion for building type classification from aerial and street view images. *Remote Sensing, 11* (11), 1259.

Hoffmann, E. J., Werner, M., & Zhu, X. X. (2020). Quality assessment of semantic tags in openstreetmap. In *, Vol. 509. IOP Conference Series: Earth and Environmental Science*. IOP Publishing.

Huang, B., Zhao, B., & Song, Y. (2018). a. Urban land-use mapping using a deep convolutional neural network with high spatial resolution multispectral remote sensing imagery. *Remote Sensing of Environment, 214*, 73–86.

Huang, G., Liu, Z., van der Maaten, L., & Weinberger, K. Q. (Jan. 2018). *Densely connected convolutional networks*, 2018b.

Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S., & Murphy, K. (Apr. 2017). *Speed/accuracy trade-offs for modern convolutional object detectors*.

Huang, R., Taubenböck, H., Mou, L., & Zhu, X. X. (2018). Classification of settlement types from tweets using lda and lstm. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium* (pp. 6408–6411). IEEE, 2018c.

Innes, J. E., & Booher, D. E. (2015). A turning point for planning theory? Overcoming dividing discourses. *Planning Theory, 14*(2), 195–213.

Kang, J., Körner, M., Wang, Y., Taubenböck, H., & Zhu, X. X. (Nov. 2018). Building instance classification using street view images. *ISPRS Journal of Photogrammetry and Remote Sensing, 145*, 44–59.

Kruspe, A., Häberle, M., Hoffmann, E. J., Rode-Hasinger, S., Abdulahhad, K., & Zhu, X. X. (2021). *Changes in twitter geolocations: Insights and suggestions for future usage*. arXiv preprint arXiv:2108.12251.

Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., Duerig, T., & Ferrari, V. (Jul. 2020). The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision, 128*(7), 1956–1981.

Langemeyer, J., Calcagni, F., & Baro, F. (2018). Mapping the intangible: Using geolocated social media data to examine landscape aesthetics. *Land Use Policy, 77*, 542–552.

Laumer, D., Lang, N., van Doorn, N., Mac Aodha, O., Perona, P., & Wegner, J. D. (2020). Geocoding of trees from street addresses and street-level images. *ISPRS Journal of Photogrammetry and Remote Sensing, 162*, 125–136.

Leichtle, T., Lakes, T., Zhu, X. X., & Taubenböck, H. (Nov. 2019). Has Dongying developed to a ghost city? - evidence from multi-temporal population estimation based on VHR remote sensing and census counts. *Computers, Environment and Urban Systems, 78*, Article 101372.

Leung, D., & Newsam, S. (2012). Exploring geotagged images for land-use classification. In *Proceedings of the ACM multimedia 2012 workshop on Geotagging and its applications in multimedia* (pp. 3–8).

Li, L.-J., Su, H., Li, F.-F., & Xing, E. (2010). *Object bank: A high-level image representation for scene classification & semantic feature sparsification*.

Liu, F., Wang, Y., Wang, F.-C., Zhang, Y.-Z., & Lin, J. (2019). Intelligent and secure content-based image retrieval for mobile users. *IEEE Access, 7*, 119209–119222.

Lopez, B. E., Magliocca, N. R., & Crooks, A. T. (2019). Challenges and opportunities of social media data for socio-environmental systems research. *Land, 8*(7), 107.

Movshovitz-Attias, Y., Yu, Q., Stumpe, M. C., Shet, V., Arnoud, S., & Yatziv, L. (2015). Ontological supervision for fine grained classification of street view storefronts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1693–1702).

Regmi, K., & Shah, M. (2019). Bridging the domain gap for ground-to-aerial image matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 470–479).

Ren, S., He, K., Girshick, R., & Sun, J. (Jan. 2016). *Faster R-CNN: Towards real-time object detection with region proposal networks.* arXiv:1506.01497 [cs].

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (Dec. 2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision, 115*(3), 211–252.

Salem, T., Workman, S., & Jacobs, N. (2020). Learning a dynamic map of visual appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 12435–12444).

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (Mar. 2019). *MobileNetV2: Inverted residuals and linear bottlenecks*, 2018b.

Shahat, E., Hyun, C. T., & Yeom, C. (2021). City digital twin potentials: A review and research agenda. *Sustainability, 13*(6), 3386.

Shi, Y., Yu, X., Campbell, D., & Li, H. (2020). Where am I looking at? joint location and orientation estimation by cross-view matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 4064–4072).

Simonyan, K., & Zisserman, A. (Apr. 2015). *Very deep convolutional networks for large-scale image recognition.* arXiv:1409.1556 [cs].

Sloan, L., Morgan, J., Housley, W., Williams, M., Edwards, A., Burnap, P., & Rana, O. (2013). Knowing the tweeters: Deriving sociologically relevant demographics from Twitter. *Sociological Research Online, 18*(3), 74–84. https://doi.org/10.5153/sro.3001

Srivastava, S., Vargas Munoz, J. E., Lobry, S., & Tuia, D. (2020). Fine-grained landuse characterization using ground-based pictures: A deep learning solution based on globally available data. *International Journal of Geographical Information Science, 34*(6), 1117–1136.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (Dec. 2015). *Rethinking the inception architecture for computer vision.* arXiv:1512.00567 [cs].

Terroso-Saenz, F., & Munoz, A. (2020). Land use discovery based on volunteer geographic information classification. *Expert Systems with Applications, 140*, Article 112892.

Vo, N. N., & Hays, J. (2016). Localizing and orienting street views using overhead imagery. In *European conference on computer vision* (pp. 494–509). Springer.

Wang, Q., Lai, J., Xu, K., Liu, W., & Lei, L. (Oct. 2018). Beauty product image retrieval based on multi-feature fusion and feature aggregation. In *Proceedings of the 26th ACM International Conference on Multimedia. MM '18* (pp. 2063–2067). New York, NY, USA: Association for Computing Machinery. arXiv:1610.02357 [cs].

Watson, V. (2009). Seeing from the south: Refocusing urban planning on the globe's central urban issues. *Urban Studies, 46*(11), 2259–2275.

Zhang, C., Sargent, I., Pan, X., Li, H., Gardiner, A., Hare, J., & Atkinson, P. M. (2019). Joint deep learning for land cover and land use classification. *Remote Sensing of Environment, 221*, 173–187.

Zhang, W., & Huang, B. (2015). Land use optimization for a rapidly urbanizing city with regard to local climate change: Shenzhen as a case study. *Journal of Urban Planning and Development, 141*(1), Article 05014007.

Zhu, Y., Deng, X., & Newsam, S. (2019). Fine-grained land use classification at the city scale using ground-level images. *IEEE Transactions on Multimedia, 21*(7), 1825–1838.