# Clustering Method for Touristic Photographic Spots Recommendation

Flavien Deseure–Charron[1[0000−1111−2222−3333]], Djebali Sonia[1[1111−2222−3333−4444]], and Guérard Guillaume[1[2222−−3333−4444−5555]]

Léonard de Vinci Pôle Universitaire, Research Center, 92 916 Paris La Défense, France
flavien.deseure--charron@edu.devinci.fr
{sonia.djebali, guillaume.guerard}@devinci.fr

**Abstract.** Tourism and photography have become very complementary, and tourists are constantly seeking the best spots to capture pictures and memorize their vacations. However, the search for the best and unforgettable photographic spots is difficult and time-consuming for tourists, especially when visiting new regions. In this paper, we propose a method for discovering tourist photo spots from geotagged photos using clustering algorithms. The clusters are characterized to determine the type of photos such as selfies or panoramic. We compare our approach to the most used clustering algorithms namely K-Means and DBSCAN. The approach is simulated and experimentally evaluated on a real photographic dataset of the French capital *Paris*. Our approach identifies the best-known, quirky and thematic spots in the reference websites.

**Keywords:** Tourism · Photographic Spots · Clustering · HDBSCAN · Knowledge Discovery.

## 1 Introduction

Nowadays, tourism is considered one of the largest and fastest-growing industries. It is a significant economic sector for many countries in the world. Tourism is deeply related to photography [4], especially because pictures allow travelers to maintain good memories of their destinations [2]. *Deborshee Gogoi* introduces in 2014 this concept as: " *Photographic tourism is that form of special interest tourism in which tourist visits a particular place with the primary aim of photographing subjects that are unique to him. The scope of photography may range from landscapes, portraits, architectures, culture, food and wildlife to even macro subjects* " [10].

With the exponential increase of compact, cheap, or user-friendly cameras, tourists tend to share more and more photographs to immortalize their experience and keep memories. This affluence of photos has led to the development of multiple photo-sharing services such as *Flickr* and *Instagram.* These platforms have redefined the way that people travel [11, 14]. According to travel websites like *Expedia*[1], the main priority for young travelers is whether their destination is visually appealing and lends itself to be photographed for posting on photo-sharing networks.

However, finding the *"best"* spots to take photographs remains a tedious task, especially when tourists do not know the region they visit [9]. The studies that have been conducted up to now are focused on the identification and discovery of a specific points that someone may find interesting using geotagged photos. These points are mentioned in literature as *hotspots*, *points of interest* (POI), and *areas of interest* (AOI) [3].

Our study focuses on the identification of the areas where the photos are captured about a POI and not the identification of POI or AOI. To distinguish those areas, they are named *Touristic Photographic Spots* (TPS) in the rest of the paper. For each TPS, a set of characteristics is determined define the kind of photos. For example, a TPS to have a panorama view, a TPS to acquire a sunset in front of a monument, etc.

This paper proposes an approach to identify the *Touristic Photographic Spots* of a POI and to qualify them. Our contributions can be resumed as the following:

- Clustering methods to determine TPS: this method take into account the various aspect of the photos such as the density of photos, the distance to the POI, the angle to the POI. In this manner, the clustering method is based both on the geographical density of photos and on the variability of the metadata. The choice of clustering algorithms must take into account both the density and the proximity of data in respectively the geographical aspect and the photographic aspect.
- A knowledge extraction to qualify each TPS: from the metadata of the cameras, we compute for each TPS some knowledge such as its popularity, the best time of the day to take photos or its focus.

This paper is organized as follows. Section 2 describes the related work for representing and discovering spots from geotagged photos. Section 3, presents our approach for identifying photographic spots using clustering methods. Section 4 describes and comments the results. Finally, Section 5 presents our conclusions and recommendations for future studies.

## 2 Related work

The studies that have been conducted up to now, in the field of tourists photography are focused on the identification of the *hotspots*, *points of interest* (POI), and *areas of interest* (AOI). As far as we know, no approach has addressed our problem which is the identification of photographic spots using geotagged photos. The closest works on our problem are the discovery of POI or AOI.

Some studies focus on *density* [6, 8]. These aggregated data by hexagons to produce density maps and therefore found the POI from the main peaks. This

[1] https://www.expedia.com, more than 750 millions monthly visitors.

approach has the advantage to be simple but the obtained density function is too smooth to be used for photo spot discovery.

Some other studies considers a quite different approach, focusing on clustering algorithms. Indeed, discovering POI from geotagged photos can be treated as a clustering problem to identify the most photographed places. More precisely, the method used for geotagged photos are part of the geospatial clustering.

*K-Means clustering algorithm* (centroid-based algorithm) is the most used one to identify clusters from geotagged data [15]. However, K-Means requires the number of clusters as an input parameter, and it detects only spherical clusters. This shape is unsuitable to reality. Indeed, peoples take photos depending on the urban structure and the topology of the region, not as a bird view.

*Density-based clustering methods* are used to identify POI because a high photos activity can be measured by density. These methods don't require the number of clusters as an input parameter. They can handle arbitrary shape clusters, and sparse regions are treated as noise. Some example of algorithms used are *Mean Shift* [5, 18], *Ordering Points To Identify the Clustering Structure* (OPTICS) [9] and *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN) [7]. Kisilevich et al. [16] propose *Photo-DBSCAN* (P-DBSCAN) a new density-based clustering algorithm based on DBSCAN, that weight the photo on various metadata. This method has been exploited and enriched during the last decade [17, 19].

Some recent studies use the topological structure of photos (spatial and non-spatial) to determine clusters such as GeoSOM based on Self-Organizing map [12]. The structure can also be studied through data mining such as FPGrowth to understand the behaviors of tourists [13]. But those methods are used to determine behaviors and patterns of people considering spatial and temporal as continuous data.

Based on these varied approaches, we can deduce that to determine *Touristic Photographic Spots*, the choice of the algorithm is important and is directly related to the data. If we take the example of the *Eiffel Tower*, we found many streets, bridges, and parks very close to the monument having diverse views and which constitute a different TPS in its own right. However, these are undetected by the most used methods seen previously. Centroid-based don't suit the urban infrastructure and Density-based may regroup excessively vast areas due to the high proximity of those areas.

Our approach will be compare to the most used clustering algorithms for POI identification namely K-Means and DBSCAN.

## 3 Our approach

Our objective is to propose a method to identify the TPS of each POI and qualify the TPS. Each POI is characterized by its type (hotel, restaurant, attraction) and localization (lat, long). The POI is photographed by several users which are identified by an identifier. Each photo is characterized by the tags of the photo, localization (lat, long), the date and time when the photo is captured. A set of data is available on the camera used for each photograph as ISO, aperture, shutter speed, and focal length.

Since we are looking for TPS as areas of various shape and size, and with a homogeneous data for the characteristics, we propose a double clustering to define TPS. The first algorithm, as a geographical clustering, will determine the shape of the areas as when the second one, a data clustering guarantee the homogeneity of the data. Then, we extract information from the clusters concerning photographic tourism.

Our approach can be resumed in three steps as follows (see figure 1):

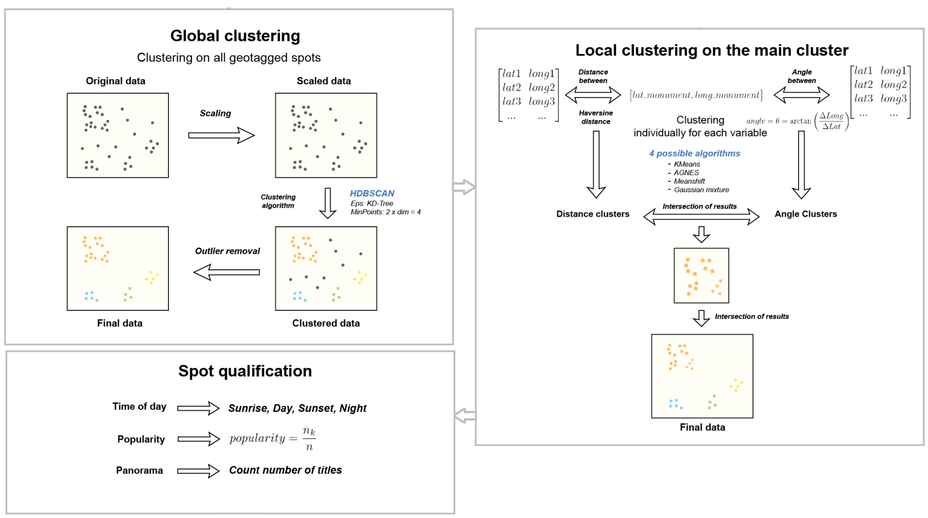



Fig. 1: Flowchart of our approach.

1. *Global clustering*: to create the continuous density areas, we apply a first clustering algorithm on the Cartesian projection associated with the photos of each POI.
2. *Local clustering*: to define TPS. For each cluster from the global clustering, the local clustering is based on two new parameters, distance to POI and angle with POI. Those greatly transform the view of any photos and are required to refine previous clusters.
3. *TPS qualification*: extraction of knowledge from photos to qualify the found TPS. The qualification is used to recommend the latter to tourists.

### 3.1 Global Clustering

The first step of our approach is to determine the shape of areas where tourists take photos. The first step for any geospatial clustering is to make a Cartesian projection of all photos. The global clustering has two main goals. Firstly, the numerous outliers has to be removed, *i.e.* the photos that cannot constitute a cluster. We define as outliers a small amount of geotagged data with a low density that are not representative of any trend. Secondly, the TPS may be at any size. It is dependent of the topology and the urban infrastructure, which constrains how photos are taken.

To achieve these two goals, we use a density-based clustering algorithm. We choose this kind of algorithms because they separate clusters by contiguous areas of low point density. The data points in the separating areas of low point density are typically considered outliers. Some existing methods like DBSCAN and OPTICS fail to identify clusters with different density levels because they are based on a "flat" (*i.e.* non-hierarchical) representation. One of the methods that solve this problem is HDBSCAN [20]. It is a clustering algorithm that extends DBSCAN by converting it into a hierarchical clustering algorithm. This method works in three steps: first, it estimates the densities around certain data to determine a threshold; then, it selects areas above this threshold density; finally, it combines points in these selected areas.

Most of the density-based methods require the assumption of a density threshold. They compute a threshold and gather the data with densities above the threshold and group theme together to form clusters. To use HDBSCAN algorithm, first, we need to estimate the density around some data to build the density landscape of the dataset. The HDBSCAN algorithm computes the *Core distance* of a random set of data thanks to the *K-th nearest neighbor* (KNN) method. Data in denser regions would have smaller Core distances while data in sparser regions would have larger Core distances. The density landscape is the inverse of the Core distance. Then, HDBSCAN builds a hierarchy to figure the right density for each cluster and how to cut like a hierarchical clustering.

In our approach we compare the HDBSCAN algorithm with the most used clustering algorithms for POI identification namely K-Means and DBSCAN.

### 3.2 Local clustering

The global clustering provides a set of clusters that contains a continuous density area of geotagged photos. Apart from the location of the photo, two other parameters as angle of view and distance to the POI may vary greatly inside each cluster. These parameters characterize the TPS differently as they alter the purpose of the photos. The angle of view and distance are greatly heterogeneous into the cluster around and near the POI, and with very large clusters. To refine the clusters, we apply a second clustering based on those two parameters.

The second clustering is not applied on all clusters but just on clusters having a threshold of variation of the angle and distance to POI. Those thresholds define the limit of acceptable heterogeneity of a cluster. The two thresholds are defined as follows:

– Clusters with a surface representing a total angle value above a threshold are refined. We fix this threshold to *one hour angle*, which corresponds to 15 degrees. It corresponds to the change of framing in photography.

– Clusters with the maximum distance between two points of the cluster must be superior to a threshold are refined. We fixed this threshold to *four-time* the *epsilon* value of the global clustering (thus two times the diameter to have a sufficient size).

To define the angle formed between all points of the cluster and the coordinates of the POI we use the following formula:

$$angle = \theta = \arctan\left(\frac{\Delta Long}{\Delta Lat}\right) \tag{1}$$

where $\Delta Long$ and $\Delta Lat$ are the difference in longitude and latitude between the data of the cluster and the POI, respectively. This angle is computed in radian.

To compute the distance between all points of the cluster and the coordinates of the POI, we use a geodesic distance *i.e.* harversine distance.

Since we cannot make assumptions about the form and density of the second clustering, we propose using various algorithms with pros and cons and to compare their results to determine which one fits these data. We used internal and external measurements to select the most appropriate method. For comparison purposes, our approach should be deterministic.

In order to choose the best algorithm, we compare the results of four algorithms with their pros and cons from the four main types of clustering:

**Partitional/centroid-based: K-Means [1]**. The K-Means algorithm is still widely used because of its simplicity and good performance.The most key parameter to set is the number of clusters. To tune this number, we use the *Elbow* method [22] using the *Within-Groups Sum of Squares* (WGSS). Indeed, we want to evaluate the cohesion of the clusters and not the separation, as the clusters are very close to each other.

**Distribution Model-Based: Gaussian Mixture Model (GMM) [1]**. This algorithm employs an interesting approach that tries to represent the dataset as a mixture of normal distribution. A GMM tends to group the data points belonging to a single distribution together. While K-Means forms spherical shape cluster, GMM can produce various ellipsoid shapes for the same dataset. For the number of components parameter, in the same way as with K-Means, we implement the Elbow method with WGSS.

**Density-Based: Mean Shift [1]**. This deterministic method update potential centroids to be the mean of the points within a given bandwidth. Mean Shift works very well on spherical-shaped data. Furthermore, it automatically selects the number of clusters contrary to other clustering algorithms like K-Means. The bandwidth was estimated by computing the *K-Nearest Neighbors* KNN algorithm as recommended.

**Hierarchical: AGNES [1]**. Hierarchical method are deterministic and constructs a hierarchical tree of distances between data, called a dendrogram. This is helpful because the algorithm produces an explicit graphical depiction of the clusters. The AGNES method is one of the most used. It adopts a bottom-up approach. For the number of clusters, we employ the Elbow method with WGSS with a *Ward linkage*.

### 3.3 Indexes and validation

After applying the global clustering and the local clustering, we intersect the results for both parameters angle and distance to produce new clusters from the main one. The evaluation is essential to our approach, as it will allow us to choose the most appropriate algorithm for the local clustering. The evaluation allows us to find the most efficient algorithm.

We implemented internal measurements for the evaluation based on the Cartesian projection of the data. The difference between algorithms can be very closed and external evaluation would be ineffective and irrelevant since we refine a single cluster. In our context, we need a high cohesion for clusters as there are not well separated. Therefore, we choose the following evaluation:

1. *Ball Hall*: it computes the mean dispersion of a cluster, *i.e.* the mean of the squared distances of the points of the cluster with respect to their center. The lower the value, the better the clustering is.
2. *Banfeld-Raftery index*: it is the weighted sum of the logarithms of the mean of the squared distances between the points in the cluster and their center. The logarithm allows smoothing of the impact of big or small clusters in the number of points. The lower the value, the better the clustering is.

The final clustering is composed of clusters from the global clustering and refined clusters from the local clustering. Since the clusters are computed from the data for a POI, they represent TPS for this POI. The process is done for each POI.

### 3.4 TPS qualification

Once the TPS of a POI are defined, we qualify these TPS to perform recommendations to tourists. We define three ways to qualify the TPS: first, according to the time of day, then, whether this TPS is a panorama or not, and finally according to the popularity of the TPS.

**Time of day**. We have broken down the day into four parts: Sunrise, Day, Sunset, Night. We use the date of the photo and the *ISO*[2] of the camera to determine the right part for each photo. We manage the time when the photo was captured with a margin of $+-10$ minutes (1.4% of the day) for sunrise and $+-20$ minutes (2.8% of the day) for sunset depending on the timezone and period of the year. We can also deduce the time of day from the *ISO* used by the photo. The higher the ISO, the more sensitive the camera sensor becomes, and the brighter your photos appear. $ISO100$ is used in a sunny and open area, $ISO400$ is used during a cloudy day, $ISO800$ and higher are used from sunset to night time of day. From each TPS, we determine the percent of each part of the days from its photos.

**Panorama**. This category indicates TPS that are likely to be panoramas. To perform this, we combined several indicators: the number of POI having this

[2] https://www.adobe.com/creativecloud/photography/discover/iso.html

TPS, the aperture, and the focal length of each photo. First, we selected all the photos located in the same cluster and counted the number of different POI taken in each photo (from the tags). We compute the mean and the ratio of photos having several POI to those with one POI. For this second indicator, the aperture value designates the width of the hole within the lens through which the light travels into the camera body. When the aperture is very narrow (below $f/8$), the depth of field is large and therefore there is a chance that the photo is a panorama. We take into account the largest aperture used among the photos and its percentage. Finally, the focal length represents the measure of the optical distance inside the lens from which all light rays converge on the image sensor of the camera. The lower the value, the wider the field of view is, and therefore the more likely it is a panorama. Focal lengths below $55mm$ are used to take large-angle photos. We take into account the largest focal length used among the photos and its percent.

**Popularity**. The third way to qualify a TPS is according to its popularity, in number of photos, and to be able to classify this TPS as unusual or unmissable for example. A direct and effective measure is the percentage of importance of the cluster as follows: $popularity = \frac{n_k}{n}$. Where $n$ remains the number of points in the dataset and $n_k$ is the number of points in the $k^{th}$ cluster.

## 4 Experiments

We conducted experiments on the social network *Flickr* over a period from 2007 to 2019. We chose Flickr because it is primarily aimed at professionals and photo enthusiasts. Moreover, it grants us access to the data related to the camera. For our case study, we have chosen the city of *Paris*, because it is one of the most attractive cities in the world; regularly ranking first among the most visited cities in the world. In our dataset gathered from Flickr focused on *Paris* region, we have $2,945,085$ geotagged photos on $1,414,816$ POIs taken by $98,555$ users with $2,948$ different camera sets.

### 4.1 Data Processing

The tags in each photo, which designate the POI taken, have not been pre-filled by the social network Flickr. Thus, it is written in many alternative ways in the dataset (misspellings, translated into other languages). To overcome this problem, we grouped all titles with a name at 85% similar *Cosine similarity* using the vector embedding generated from *Sentence-Bert* [21]. At that time, we removed photos captured in a short time interval by the same user. As we want to extract knowledge from our clusters, especially at the time of day when a POI was taken, we chose a duration of five hours at minimum between two photos. This allows us to take into account the users who took the POI during the day and at sunset for example. Then, for determining the coordinates of the POI, we use the *Open Topo Data API* [3]. Finally, we project the data into a Cartesian plane to perform our method.

To experiment with our approach, we present the result of the *Eiffel Tower* POI. We choose this POI because, with more than 7 million visitors a year, it is the most visited and photographed monument in the world on various social networks like Flickr and Instagram. Since the monument is seen on most websites about *Paris* and lots of media provides good analytics of the photographic tourism of the *Eiffel Tower*, we can easily compare our results to them. In this specific POI, after pre-processing, we get 17,781 photos taken by 11,372 users with 674 different cameras.

To validate our approach, we compare the results with the spots referenced on the blogs and touristic websites: lonelyplanet with 3.2 millions monthly visitors[4], lodgisblog with, 90 thousands monthly visitors (french), blog specialized in photographic tourism[5], image banks (Google Images, Instagram, Pinterest).

### 4.2 Global clustering comparison

Based on the 17,781 photos tagged *Eiffel Tower* we produce a Cartesian projection then we implement the clustering algorithm. We initially performed the global clustering using three methods: HDBSCAN, KMeans, and DBSCAN.

**Conventional methods.** First of all, K-Means has been chosen as a reference as it is one of the most used and simple clustering methods. This method randomly initiates $K$ points in the data as centroids and assigns all points to the nearest centroids. Then the centroid moves to the average of the points assigned to it. And we rehearse this step until convergence. The most critical parameter to determine is the number of clusters. To acquire the most proper value for this number, we compute the algorithm in a range from 2 to 100 and compute two metrics: *Silhouette* and *Davies-Bouldain*. The objective is to determine the value that maximizes the Silhouette score and minimizes the Davies-Bouldain score. Finally, we compute the mean value rounded up to the nearest whole number between the best number of clusters according to each index and employ it as a parameter for our method.

Secondly, DBSCAN has been chosen as the most used density-based method. For the *epsilon* value estimation, we use the *k-dimensional tree* (kd tree) method: we compute the Nearest Neighbors algorithm on our data and get the distances between all neighbors. Then, we select the elbow of the curve of the distance and use the associated distance as the epsilon point. For the *minPoints* parameter, we set the value to 4, which is equivalent to $2 * number\_of\_dimension$ as recommended in the original paper of DBSCAN [7]. Finally, we choose the Euclidean distance for performance reasons.

[3] https://www.opentopodata.org/
[4] https://www.lonelyplanet.fr/article/10-points-de-vue-sur-la-tour-eiffel
[5] http://blog.lodgis.com/top-10-des-vues-sur-la-tour-eiffel/

**Indexes and validation of the global clustering.** To determine the best clustering between K-Means, DBSCAN, and HDBSCAN, we will use the following indexes:

1. Banfeld-Raftery: As reminder, it measures the mean of the squared distances between the points of a cluster and its center. The lower the value, the more points are closed to the center.
2. Davies-Bouldain: The score is the average of the maximum ratio between the distance of a point to the center of its cluster and the distance between two clusters centers. As a result, clusters that are farther apart and less scattered will score higher. The minimum value for this index is 0 and lower values indicate better clustering.
3. Calinski-Harabasz: It is the ratio between the between-cluster variance and the within-cluster variance. The Calinski-Harabasz index varies between 0 (worst clustering) and $infinite$ value (best clustering). It increases linearly with the number of points in the sample. Therefore, its order of magnitude can vary considerably from one dataset to another.

From those indexes, we choose the clustering with the highest Banfeld-Raftery value, the lower Davies-Boudlain index, and the highest Calinski-Harabasz index.

**Results** As a reminder, the cohesion of the clusters is the most important aspect as opposed to the separation. Indeed, two photos spots can be very close or even touch each other and thus decrease the separation value and all associated measurements. The Table 1 shows the results of the three clustering algorithms and the values of the cohesion index *Banfeld-Raftery*, *Davies-Bouldain* and *Calinski-Harabasz* of each algorithm.

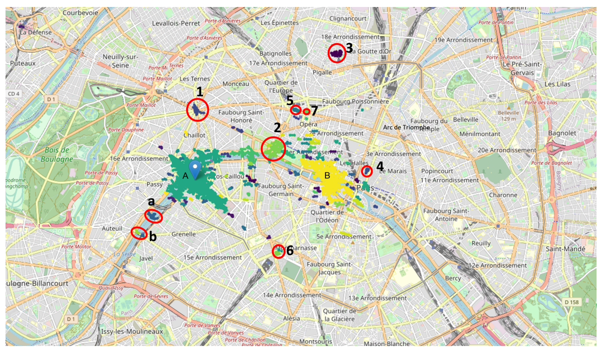
Table 1: Internal measurements comparison of classical methods (Eiffel Tower).

| | K-Means | DBSCAN | HDBSCAN |
|---|---|---|---|
| Banfeld-Raftery | $-1.86e^5$ | $-1.88e^5$ | $-1.94e^5$ |
| Davies-Bouldain | $7.3e^{-1}$ | $5.06e^{-1}$ | $5e^{-1}$ |
| Calinski-Harabasz | $1.3e^4$ | $4.39e^3$ | $4.4e^4$ |

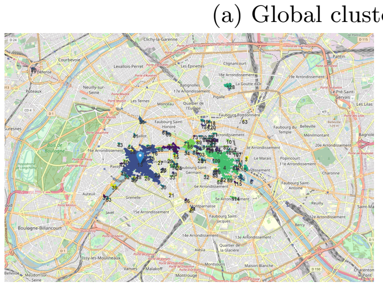
As a reminder, we choose the clustering with the highest Banfeld-Raftery value, the lower Davies-Boudlain index, and the highest Calinski-Harabasz index. HDBSCAN presents the best overall results, close to DBSCAN. Thus, we choose its cluster with $-1.94e^5$ Banfeld-Raftery value, $5e^{-1}$ Davies-Boudlain index and $4.4e^4$ Calinski-Harabasz index. The figure 2 presents the results of the three algorithms. K-Means (figure 2c) provides clearly some unintelligible clusters. The main difference between HDBSCAN (figure 2a) and DBSCAN (figure 2b) is the clusters of small size are not discovered by DBSCAN. Moreover, since

DBSCAN have no regulation on the density/sparsity of data, it tends to find compact cluster which affects clusters or borders with less density than the fixed threshold of DBSCAN.

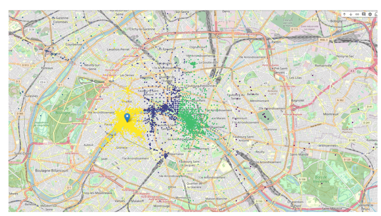
The global clustering also allowed identifying the main photographic spots (figure 2a). We noticed that most of photographic spots use on the internet were identified with our method. Most of the spots are widely shown is the reference websites about the *Eiffel Tower* : 1) *Arc de Triomphe*, 2) *Place de la Concorde*, 3) *Montmartre*, 4) *Centre Pompidou*, 5) *Printemps Haussman* (rooftop), 6) *Tour Montparnasse*, 7) *Galeries Lafayette* (rooftop). The global clustering also allowed to identify of photographic spots not present on the reference sites like the two bridges a) *Grenelle bridge*, b) *Mirabeau bridge*, etc.




(a) Global clustering with HDBSCAN.



(b) Global clustering with DBSCAN.



(c) Global clustering with K-Means.

Fig. 2: Global clustering comparison.

### 4.3 Local clustering comparison

Two clusters are selected for the local clustering: $A$ (in dark green) and $B$ (in yellow) from figure 2a. In the results, we present the local clustering for the first selected cluster.

We test the four algorithms: K-Means, Mean Shift, DBSCAN, Agglomerative clustering. We compare results with Ball Hall and Benfeld-Raftery index (table.

2). As a reminder, we choose the clustering with the lowest Ball Hall value and Benfeld-Raftery index.

Table 2: Internal measurements comparison of algorithms used for local clustering (Eiffel Tower).

| | KMeans | AGNES | Mean Shift | Gaussian Mixtures |
|---|---|---|---|---|
| Ball Hall | $8e^{-5}$ | $8e^{-5}$ | $1e^{-4}$ | $8e^{-6}$ |
| Banfeld-Raftery | $-1.65e^{5}$ | $-1.60e^{5}$ | $-1.67e^{5}$ | $-1.66e^{5}$ |

The results of the algorithms are presented in Fig. 3. The GMM clustering provides the best results which are also close to the urban structure near the *Eiffel Tower*. From top left to bottom right, we can mention clusters located at *Trocadero's Garden*, *Alma's bridge* with *Palais de Tokyo*, *Iéna's bridge* with *Carrousel*, closest northern roads (*Quai Branly*), closest eastern roads (*Bourdonnais avenue and University*), the inner ring represents the various points of view at the feet of the tower, then on the left in light blue we got *Bir Hakeim bridge*, following by the *Emile Antoine stadium*, the *Hotel Pullman* and finally the bottom pink cluster represents the *Champ de Mars*.

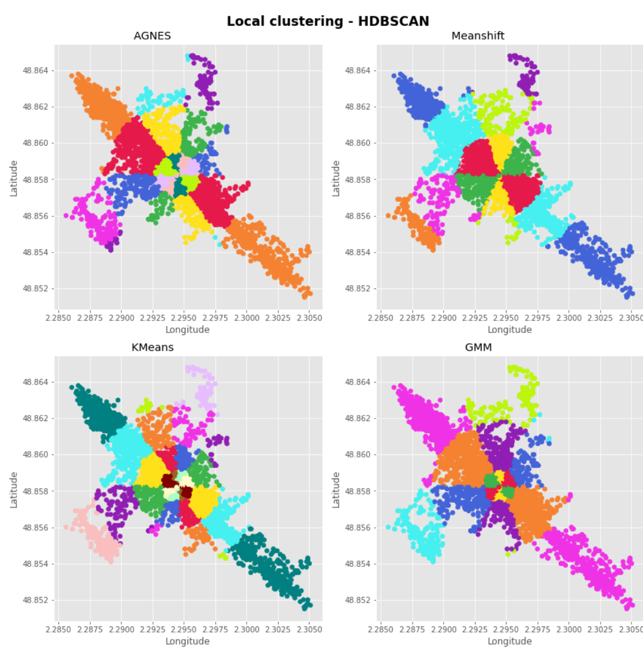



Fig. 3: Comparison of the four algorithms for local clustering on the Eiffel Tower in Paris.

As shown, the local clustering provides a more comprehensive view of photographic tourism near the $Eiffel\ Tower$. The various cited places offer various points of view and ways to handle the tower's environment.

### 4.4 Spot qualification

*Time of day* . During our experiments, we discovered good spots for sun dusk as the proportion differs a lot from the reference one (between 7% and 10% of all photos TPS instead of 2.78%), see TPS in Figure 4. Those spots are on the east, northeast of the $Eiffel\ Tower$, and they are at a higher altitude. Some spots are close to the POI, tourists capture the sun dusk between the feet of the $Eiffel\ Tower$. One TPS on the bottom left is the *Grenelle bridge* (number 26), and many others are on various bridges. Tourists can see the sun dusk glare on the water in front of the tower. We conclude those spots are ideal to take the sun dusk close to the $Eiffel\ Tower$.

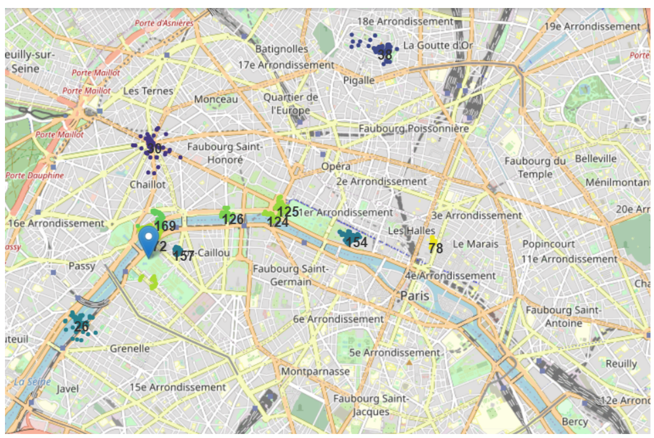



Fig. 4: Clusters with a high percent of sun dusk.

*Panorama* . Concerning the panorama, we obtained the following map Fig. 5). We found most of panorama's like *La Défense* (number 1), *Montmartre* (number 2), *Arc de Triomphe* (number 3), some bridges, the *Place de la Concorde* (number 4), *Saint Honoré district* (number 5) and *Les Halles* (number 6). Those places offer an advantageous point of view of some parts of Paris and are at higher altitudes or with vast open views. Those results are biased as most tourists want to place the $Eiffel\ Tower$ in every photo.

*Popularity* . The most popular spots are all around the monument (main global cluster) and in large places (*Place de la Concorde*, *Montmartre*, *Arc de Triomphe*). Moreover, most of the photos on the internet are taken from those places.

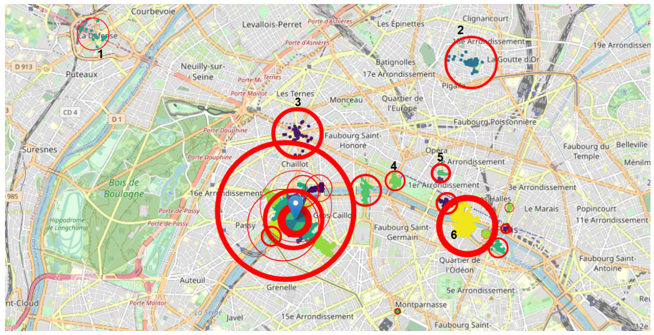



Fig. 5: Panorama's detection for Eiffel Tower

A deep discussion about our results is presented in our github[6].

## 5 Conclusion and Future work

Our approach allows solving the problem of identifying Touristic Photographic Spots. It has succeeded in addressing the problem of places "hidden" by the main cluster using a double clustering approach. Our context-free method adapts itself to the dataset. It determines TPS with characteristics thanks to a benchmark of adapted methods and index comparison.

In future works, we will enhance the spot's qualification. Flickr grants us access to the *EXIF* data of the images. It is therefore possible to qualify the spots by using the photo and camera metadata. Moreover, we need to strengthen our methods. We are also considering adopting the Self-Organizing Map, Spectral Clustering and Spacial Clustering with the following five dimensions to find TPS: three spatial dimensions, distances to POI, and angles with POI.

## References


1. Benabdellah, A.C., Benghabrit, A., Bouhaddou, I.: A survey of clustering algorithms for an industrial context. Procedia computer science **148**, 291–302 (2019)
2. Berger, H., Denk, M., Dittenbach, M., Pesenhofer, A., Merkl, D.: Photo-based user profiling for tourism recommender systems. In: EC-Web. pp. 46–55. Springer (2007)
3. Camilleri, M.A.: The tourism industry: An overview. Travel marketing, tourism economics and the airline product pp. 3–27 (2018)
4. Cederholm, E.A.: The use of photo-elicitation in tourism research–framing the backpacker experience. Scandinavian Journal of Hospitality and Tourism **4**(3), 225–241 (2004)


[6] https://github.com/flaviendeseure/Clustering-Method-for-Touristic-Photographic-Spots-Recommendation.git

5. Clements, M., Serdyukov, P., De Vries, A.P., Reinders, M.J.: Using flickr geotags to predict user travel behaviour. In: Proceedings of the 33rd international ACM SIGIR conference. pp. 851–852 (2010)
6. Da Rugna, J., Chareyron, G., Branchet, B.: Tourist behavior analysis through geotagged photographies: a method to identify the country of origin. In: 2012 IEEE 13th International Symposium CINTI. pp. 347–351. IEEE (2012)
7. Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: kdd. vol. 96, pp. 226–231 (1996)
8. García-Palomares, J.C., Gutiérrez, J., Mínguez, C.: Identification of tourist hot spots based on social networks: A comparative analysis of european metropolises using photo-sharing services and gis. Applied Geography **63**, 408–417 (2015)
9. Gavric, K.D., Culibrk, D.R., Lugonja, P.I., Mirkovic, M.R., Crnojevic, V.S.: Detecting attractive locations and tourists' dynamics using geo-referenced images. In: 2011 10th, TELSIKS. vol. 1, pp. 208–211. IEEE (2011)
10. Gogoi, D.: A conceptual framework of photographic tourism. IMPACT: IJRANSS **2**, 109–114 (2014)
11. Gretzel, U.: Tourism and social media, vol. 2. Sage Thousand Oaks, CA (2018)
12. Henriques, R., Bacao, F., Lobo, V.: Exploratory geospatial data analysis using the geosom suite. Computers, Environment and Urban Systems **36**(3), 218–232 (2012)
13. Höpken, W., Müller, M., Fuchs, M., Lexhagen, M.: Flickr data for analysing tourists' spatial behaviour and movement patterns: A comparison of clustering techniques. Journal of Hospitality and Tourism Technology (2020)
14. Kang, M., Schuett, M.A.: Determinants of sharing travel experiences in social media. Journal of Travel & Tourism Marketing **30**(1-2), 93–107 (2013)
15. Kennedy, L., Naaman, M., Ahern, S., Nair, R., Rattenbury, T.: How flickr helps us make sense of the world: context and content in community-contributed media collections. In: Proceedings of the 15th ACM Multimedia. pp. 631–640 (2007)
16. Kisilevich, S., Mansmann, F., Keim, D.: P-dbscan: A density based clustering algorithm for exploration and analysis of attractive areas using collections of geo-tagged photos. In: Proceedings of the 1st COM-GEO international conference. pp. 1–4 (2010)
17. Kuo, C.L., Chan, T.C., Fan, I., Zipf, A., et al.: Efficient method for poi/roi discovery using flickr geotagged photos. ISPRS International Journal of Geo-Information **7**(3), 121 (2018)
18. Lu, X., Wang, C., Yang, J.M., Pang, Y., Zhang, L.: Photo2trip: generating travel routes from geo-tagged photos for trip planning. In: Proceedings of the 18th ACM international conference on Multimedia. pp. 143–152 (2010)
19. Majid, A., Chen, L., Chen, G., Mirza, H.T., Hussain, I., Woodward, J.: A context-aware personalized travel recommendation system based on geotagged social media data mining. International Journal of Geographical Information Science **27**(4), 662–684 (2013)
20. McInnes, L., Healy, J., Astels, S.: hdbscan: Hierarchical density based clustering. J. Open Source Softw. **2**(11), 205 (2017)
21. Reimers, N., Gurevych, I.: Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084 (2019)
22. Satopaa, V., Albrecht, J., Irwin, D., Raghavan, B.: Finding a" kneedle" in a haystack: Detecting knee points in system behavior. In: 2011 31st ICDCS. pp. 166–171. IEEE (2011)