Multimodal Technologies and Interaction

MDPI

Article

Text Spotting towards Perceptually Aliased Urban Place Recognition

Dulmini Hettiarachchi [1,*], Ye Tian [1], Han Yu [1] and Shunsuke Kamijo [2]

[1] Graduate School of Interdisciplinary Information Studies (GSII), The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan
[2] The Institute of Industrial Science (IIS), The University of Tokyo, 4 Chome-6-1 Komaba, Meguro City, Tokyo 153-0041, Japan
* Correspondence: dulmini@kmj.iis.u-tokyo.ac.jp

Abstract: Recognizing places of interest (POIs) can be challenging for humans, especially in foreign environments. In this study, we leverage smartphone sensors (i.e., camera, GPS) and deep learning algorithms to propose an intelligent solution to recognize POIs in an urban environment. Recent studies have approached landmark recognition as an image retrieval problem. However, visual similarity alone is not robust against challenging conditions such as extreme appearance variance and perceptual aliasing in urban environments. To this end, we propose to fuse visual, textual, and positioning information. Our contributions are as follows. Firstly, we propose VPR through text reading pipeline (VPRText) that uses off-the-shelf text spotting algorithms for word spotting followed by layout analysis and text similarity search modules. Secondly, we propose a hierarchical architecture that combines VPRText and image retrieval. Thirdly, we perform a comprehensive empirical study on the applicability of state-of-the-art text spotting methods for the VPR task. Additionally, we introduce a challenging purpose-built urban dataset for VPR evaluation. The proposed VPR architecture achieves a superior performance overall, especially in challenging conditions (i.e., perceptually aliased and illuminated environments).

Keywords: visual place recognition; urban place recognition; text spotting; image retrieval; visual search

Citation: Hettiarachchi, D.; Tian, Y.; Yu, H.; Kamijo, S. Text Spotting towards Perceptually Aliased Urban Place Recognition. *Multimodal Technol. Interact.* **2022**, *6*, 102. https://doi.org/10.3390/mti6110102

Academic Editor: Heysem Kaya

Received: 25 October 2022
Accepted: 15 November 2022
Published: 18 November 2022

1. Introduction

We struggle to recognize places in foreign environments due to unfamiliarity or inability to understand the surrounding language. Modern-day smartphones equipped with integrated sensors (i.e., camera, GPS) provide opportunities to create intelligent solutions to aid. There is also an increased demand for such new services [1].

Recognizing places using visual information is studied as Visual Place Recognition (VPR). It is widely studied for localization and SLAM tasks which demand high precision [1,2]. In contrast, this work aims to recognize urban places of interest (POI) depicted on an image as a visual search task and there has been a scarce focus on it. Landmark recognition is a closely related task and a number of studies [3–6] have adopted an image retrieval approach. However, in addition to landmarks that display distinctive features, urban POIs include business entities (e.g., shops, restaurants) and commercial buildings (e.g., shopping malls, offices). Additionally, urban environments go through frequent appearance variances (i.e., structural, seasonal, and illumination changes) and display perceptual aliasing (i.e., two distinct places appearing visually similar) (Figure 1). Therefore, visual similarity alone is not robust against these challenging conditions [7].

Appearance invariant descriptors [8,9], a continuously growing database [10,11], and image transformation [12,13] are proposed as solutions to overcome the appearance variant problem in visual localization tasks. However, these methods demand additional storage and processing, which is unfavorable to real-world applications.

Figure 1. Examples of urban VPR challenges: (**a**) Appearance variance—daytime vs. illuminated; (**b**) Appearance variance—seasonal; (**c**) Perceptual aliasing—shops at the same building complex; (**d**) Perceptual aliasing—nearby closed shops with metal roll-down gates.

Humans leverage texts available in the environment for place recognition. Texts remain distinct for perceptually aliased entities and unchanged under appearance variance. Few studies have used text detectors to create visual descriptors [14–16] for mobile robot localization tasks. However, due to the difference in task requirement (high precision localization vs. place recognition), these methods that generate textual descriptors and topological maps are unnecessarily expensive for the urban place recognition task. To the best of our knowledge, no studies have exploited the use of scene text spotters (i.e., end-to-end (E2E) text detection and recognition) for the urban place recognition task.

To this end, we propose a novel VPR through text reading pipeline (VPRText) and then propose a hierarchical VPR architecture (VPRTextImage) that combines VPRText and image retrieval. We directly use the transcript produced by off-the-shelf text spotting algorithms to recognize the place, making the process inexpensive by eliminating the need to extract and store textual descriptors. Place text identification with layout analysis and text similarity search modules are proposed to address challenges (i.e., varied layouts, partial readings, name discrepancies) in place recognition through text reading (Figure 2). To the extent of our knowledge, no previous studies have proposed a pipeline that fuses visual, textual and, positioning information for the place recognition task.

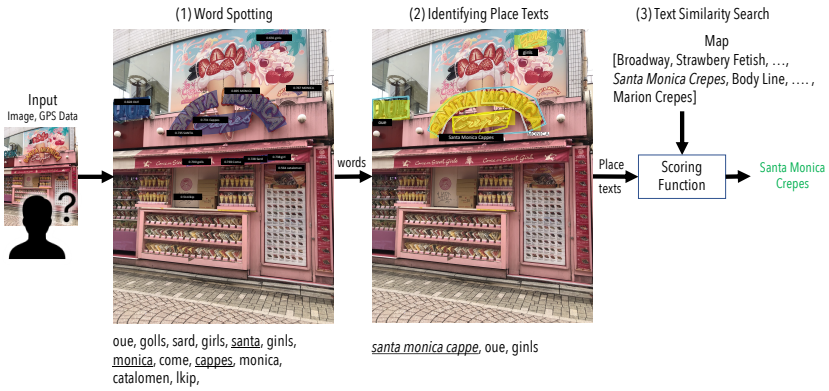

Figure 2. An example of place recognition through text reading: (1) detection and recognition of all word instances (oue, golls, sard, girls, santa, ginls, monica, come, cappes, monica, catalomen, lkip); (2) performing layout analysis (i.e., words into lines/regions) and filtering irrelevant texts (e.g., menus, banners) to identify the full place display texts (santa monica cappe, oue, ginls); (3) matching display texts against the listed place names to obtain the top match.

Our contributions are (1) propose a VPR through text reading pipeline (VPRText) that can accommodate off-the-shelf text spotting algorithms and perform layout analysis and text similarity search to recognize places using their display names, (2) present VPRTex-

tImage, an architecture that combines VPRText and image retrieval, complemented with positioning information to tackle challenging conditions such as appearance variance and perceptual aliasing, (3) perform a comprehensive empirical study on the applicability of SoA text spotting, detection, and recognition algorithms for the VPR task, (4) evaluate on a new purpose-built urban place dataset with challenging text instances, environmental conditions and, perceptual aliasing. We compare VPRText against the image retrieval approach on our dataset and demonstrate that VPRText outperforms, especially under challenging conditions. Through comprehensive analysis, we discuss the limitations of different approaches to VPR and demonstrate that VPRTextImage can perform well under challenging conditions as well as textless environments.

The rest of the paper is organized as follows. Section 2 reviews related work followed by methodology in Section 3. Experimental results are given in Section 4, and the conclusion is drawn in Section 5.

2. Related Work

2.1. Visual Place Recognition

Visual Place Recognition (VPR), recognizing the place depicted on a given image, has attracted multiple scientific communities including computer vision, robotics, and machine learning. It is studied in different domains and can be categorized based on the three key drivers; (i) Agent (e.g., autonomous cars, mobile robots, aerial vehicles), (ii) Environment (indoor vs. outdoor, structured vs. open, artificial vs. natural), and (iii) Downstream task (e.g., localization, SLAM, navigation). These drivers impose the problem definition, solution design, and evaluation of the spatial artificial system [2]. The majority of the studies focus on localization and SLAM tasks targeted at mobile robots or autonomous vehicles [2]. These tasks aim to determine the accurate location or camera pose with respect to the environment and demand high precision [1]. In contrast, this study focuses on the identification of the place and is similar to a visual search task. Therefore, it demands high recall over precision.

Landmark recognition can be considered a sub-task of POI recognition. It is successfully approached as an image retrieval problem, retrieving the most similar image to the given image, from an image database [3–6]. Recent success in landmark recognition attributes to deep-learned global and local descriptors [1–4,17]. With the advancement of deep learning and the introduction of large-scale datasets, it is also approached as an instance recognition task, casting as an extreme classification problem [4–6]. However, in addition to landmarks, the urban POIs include business entities (e.g., shops, restaurants, cafes) and commercial buildings (e.g., shopping malls, office complexes). Urban environments go through frequent appearance variances (i.e., structural, seasonal, and illumination changes) and display perceptual aliasing (i.e., two distinct places appearing visually similar). As image retrieval is based on image similarity, it may believe visually similar distinct places to be the same and visually different instances of the same place to be different. The instance recognition approach questions the extensibility of the system for a real-world task, as the addition of a new place may require further training.

Studies have proposed several solutions for the extreme appearance variance challenge. Solutions include deep-learned robust appearance invariant descriptors [8,9], continuing to grow the database with newly captured images under different conditions [10,11], using image transformation techniques to replace the query image with an appearance-transformed synthetic image to align with the database condition [12,13] and supplementing the pipeline with 3D models [18]. However, these solutions demand additional storage and processing requirements (i.e., transforming images, constructing 3D models, continuous expansion of database), which may not be favorable towards real-world applications.

A limited number of studies have used text information to overcome challenging conditions for visual localization tasks. TextSLAM [15] uses text-level semantic information to obtain coarse global localization and then obtain fine local localization using the Monte Carlo localization (MCL) method based on laser data for mobile robots. Textplace [14]

generates a textual descriptor (set of text strings and their bounding box positions) to represent each image frame. Then, build a topological map with each node representing the image with textual descriptor and camera pose. They achieve place recognition by matching textual descriptors, followed by localization by modeling the temporal dependence of a camera and its motion estimation. Several other studies have also proposed the use of text information for the downstream task of robot localization [16,19] and valet parking [20]. These studies leverage text information to overcome challenging conditions, such as appearance variance and perceptual aliasing. However, their pipelines are intended for localization and are unnecessarily complex and expensive for our task of interest.

In contrast, we read text instances on a given query image and directly use the transcript to search a place directory to find the place. A place directory stores georeferencing by place name or identifier, in contrast to geotagged image databases. We propose scene text layout analysis and text similarity search modules to tackle place retrieval challenges. We then propose to fuse VPR through text reading and image retrieval to support textless environments. This is a different approach from the textual descriptor-based localization and pure image retrieval-based VPR. To the best of our knowledge, no previous studies have fused visual, positioning, and textual information for the urban place recognition task.

2.2. Scene Text Detection, Recognition, and Spotting

Scene text detection and recognition is a topic that has been studied for decades. Methods before the deep learning era mainly extracted handcrafted image features and performed repetitive processing. Whereas, recent methods utilize deep learning-based models that benefit from automatic feature learning. Recent scene-text detection, recognition, and spotting methods have focused on more challenging aspects such as arbitrary-shaped, multi-lingual, and street-view texts [21,22]. Thus, these increase the applicability to real-world applications.

Scene-text detection is the task of detecting and localizing the text instances on an image. Text detectors initially followed a multistep process, predicting local segments before grouping them into text instances [23]. Later, methods inspired by general object detectors directly predict words or text lines [24]. However, text in the wild pose more specific challenges such as oriented, curved, and vertical-shaped text. Therefore, more recent methods focus on addressing these challenges by proposing text detectors that can handle arbitrary shapes using segmentation and sub-component level methods [25–27].

Scene-text recognition is the task of transcribing the detected text into linguistic characters. Text recognizers are broadly categorized as CTC (Connectionist Temporal Classification) based methods [28] and encoder-decoder methods [29–32]. Rectification modules (e.g., TPS [33], STN [34]) are adapted to handle irregular texts. Other techniques such as the use of language models, lexicons, and semantic information are also used to further boost performance.

Recently, there has been a surge of interest in building E2E models that perform text detection and recognition. The unified task is known as text spotting. Earlier text spotting frameworks used independent detection and recognition modules sequentially. Then, text spotting frameworks were designed in a way to share the convolutional features among the two detection and recognition branches. Recent works have proposed E2E trainable frameworks to perform detection and recognition in a more unified way [35–37]. MANGO [36] proposes a Mask AttentioN Guided One-stage text spotting, in which character sequences can be directly recognized without a RoI (region of interest) operation. ABCNet [35] proposes a parameterized Bezier curve adaptation for text detection and a BezierAlign layer for feature extraction. Text Perceptron [37] proposes a segmentation-based detector and a novel shape transformation module to handle arbitrary shapes.

Recently researchers are approaching the problem from a diversity of perspectives, trying to solve different challenges including arbitrary-shaped text [38], multi-lingual scene text [39], and street view text [40]. Even though scene-text spotting is rapidly growing, the generalization ability (training on one set and evaluation on another) and the

adaptability to varying environments are less explored [21]. In this study, we evaluate the generalization and task adaptability of scene-text spotters.

Scene Text spotting has many use cases in real-world applications. It is leveraged for translations, scene understanding for autonomous vehicles, localization, navigation, and image retrieval. A limited number of studies have leveraged detection and recognition methods for visual localization tasks [14–16,19]. The study [41] evaluates the ability of a few scene-text detection and recognition algorithms in reading street view text for real-world intelligent transportation systems using a Chinese street view dataset. In this study, we evaluate the applicability of two-step spotters (detectors and recognizers) and E2E spotters for English street-view text reading and place recognition tasks. To the best of our knowledge, no studies have evaluated the SoA E2E text spotting on real-world place recognition tasks.

2.3. Datasets

Numerous datasets are introduced focusing on different domain tasks, agents, and environments [1,2,42]. These datasets can mainly be categorized as landmark recognition datasets and urban VPR datasets.

There are several urban VPR datasets [14,43] available. However, the desired task of these datasets is to localize a given view (geo-localization and/or camera pose estimation). For that purpose, datasets comprise sequences of images from selected streets or images taken at distinct locations to depict the surrounding environment from different viewpoints, rather than focusing on specific POI(s).

Landmark datasets [4,6] focus on recognizing the landmark depicted on a given image, which is related similarly to our task. Therefore, these datasets contain images focusing on landmarks. However, these landmarks do not usually contain visible text and are distinguishable by their distinctive features. In contrast, this study focus on urban POI(s), including shops, restaurants, commercial buildings, etc., which often contain textual information for identification.

Therefore, these datasets do not provide a fair setting for the evaluation of our study. Additionally, most of the existing VPR datasets comprise query and reference image sets/sequences with similar images/image sequences as the ground truth. However, the approach of our study is to achieve place recognition through text reading (eliminate the need for image retrieval) when place text signs are available. To evaluate our algorithm, we need a place directory (database where georeferencing is stored by place name or id), with place identifier as the ground truth.

Various benchmark datasets are introduced to evaluate different challenging aspects of text detection and recognition algorithms. Large-scale street view [44] dataset evaluate the street sign reading and is partly related to our task. However, the dataset is limited to Chinese street signs and the benchmark does not evaluate the VPR task.

Existing datasets limit the implementation and evaluation of the proposed algorithm. Therefore, we introduce a new dataset with a place directory for VPR through text reading and a place image database for VPR through image retrieval, to evaluate and compare our proposed algorithm.

3. Methodology

The study aims to propose a VPR architecture (Figure 3) that is robust under challenging conditions (e.g., appearance variance, perceptual aliasing). To this end, we leverage visual, positioning, and textual information. The proposed architecture first attempts to recognize the POI through text reading and, if unsuccessful, falls back to an image retrieval approach. The positioning information is used to narrow down the search space.

We approach this systematically. First, the applicability of SoA text spotters in reading place name signs is evaluated. Then, a generic pipeline for VPR through text reading (VPRText) is proposed, and finally, a hierarchical VPR architecture (VPRTextImage) com-

bining VPRText and image retrieval is proposed. In this section, we present the process and motivation in detail.

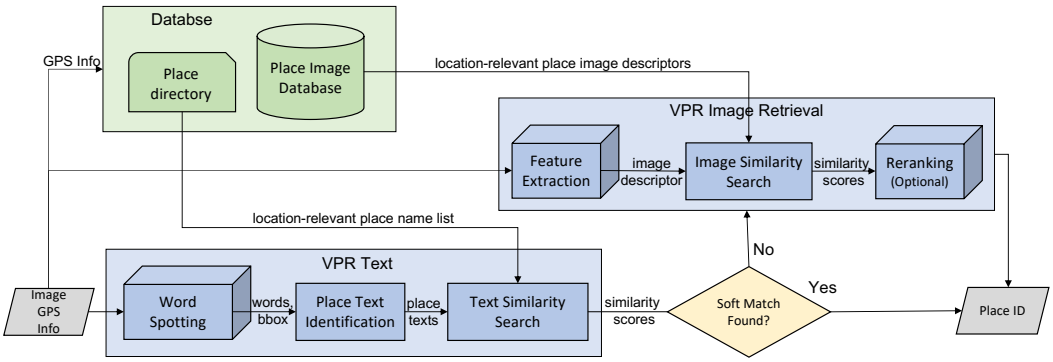

Figure 3. Proposed VPRTextImage architecture, comprising both text spotting and image retrieval modules. Given the query image and approximate location, VPR through text reading is attempted, if unsuccessful (i.e., no matching place names or no text detections) VPR through image-retrieval is attempted. Positioning information is used to narrow down the retrieval space. (In the diagram, cuboids represent the usage of deep learning-based algorithms).

3.1. Problem Definition

As the first step, through a thorough literature study and an empirical study, the challenges and, requirements for VPRText are identified. Place name signs include varied font styles, sizes, arbitrary shapes (e.g., horizontal, vertical, and curved), languages, layouts, special characters (e.g., #,-,&), numbers, and accented characters (e.g., č,è). As discussed in Section 2, ongoing research in scene-text reading attempts to tackle many of these challenges.

Recognizing place texts in their occurring layout is a unique challenge for place sign reading. Scene-text spotters and detectors commonly detect text instances as separate words. However, in natural setting, place names appear as single words, lines, or multiple lines (regions). For example, as depicted in Figure 2, the words—'santa', 'crepe' and 'monica' are separately detected, whereas the actual place name is "Santa Monica Crepes". Furthermore, cluttered backgrounds and the high availability of various irrelevant text instances, including product boards, price signs, banners, billboards, etc., create noise, making it challenging to identify the place name sign. Moreover, scene-text spotting and detecting algorithms output various bounding shapes, including quadrilaterals, polygons, or beziers. Hence, a generic pipeline should support these shapes.

Therefore, based on these observations, when proposing a generic pipeline for VPR through text reading, we have identified the following requirements:

1. Ability to identify the place name texts (i.e., filter noise);
2. Support varied layouts (i.e., words, lines, or regions);
3. Support a variety of output bounding shapes.

On the surface level, matching the recognized place text against the listed place names to obtain the place identifier may sound trivial. However, we observe the following challenges hindering the results. Refer to Figure 4 for examples.

- Partial detections caused by occlusion (e.g., Velt gelato vs Eisewelt Gelato);
- incompetency of the spotter (e.g., candy vs. candy a go go!);
- Inaccurate readings (e.g., allbirds vs albirols);
- Discrepancies between the display name and listed name (e.g., Noa Coffee vs. Noa Cafe Harajuku);
- Commonly used words leading to incorrect matches (e.g., cafe, salon, Harajuku);
- Mismatching spaces (e.g., Gustyiestudio vs. Gu Style Studio Harajuku);
- Repetitive words among texts (e.g., Body Line, LINE, Body Shop);
- Occurrences of symbols (e.g., -,&, #, emojis);
- Accented characters (ć,è);

- Separate identification of place name and tag-line.

Therefore, we aim to propose a scoring method to perform text search while tackling these challenges in retrieving the top matching listed place name.

Figure 4. Examples of observed place retrieval challenges: (**a**) Partial recognition due to occlusion; (**b**) Partial detection by the text spotter; (**c**) Occurrence of symbols; (**d**) Mismatching spacing; (**e**) Challenge in separating place name and tag line; (**f**) Occurrence of common words (Coffee) and discrepancies between display name and listed name. (Recognized display text and listed place name is given below each image).

3.2. VPRText Pipeline

In this section, we present the details of the VPR through text reading pipeline, referred to as VPRText (Figure 5). VPRText accepts a query image containing positioning information (longitude, latitude, and positioning error). First, it performs word spotting to obtain all visible texts. Then, these instances are further processed to identify the place text candidates. Finally, the recognized place texts are matched against a location-relevant place name list from a directory to retrieve top K place recognition results.

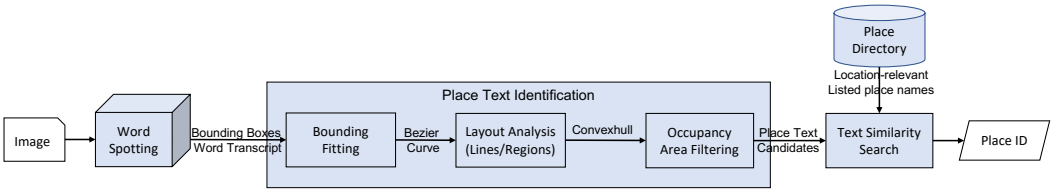

Figure 5. The proposed VPR through text reading pipeline (VPRText). Given a query image, the pipeline goes through 3 main stages to output the place identifier. (1) Word spotting—detection and recognition of all word instances; (2) Place text identification—(i) transform bounding shapes into bezier curve fitting, (ii) performs layout analysis (i.e., words into lines/regions appropriately), and (iii) removal of irrelevant words by occupancy area (e.g., product boards, banners, etc.) to obtain display place texts candidates; (3) Text similarity search—performs iterative similarity scoring between place text candidates and listed place names to obtain the top matching place identifier.

3.2.1. Word Spotting

Word Spotting can be achieved using a unified E2E text spotter [35–37,45] or a two-step method by combining independent text detection [26,27], and recognition [29–31] modules. Based on our experimental results, we propose to use an the E2E text spotters. Text spotting algorithms accept a query image and output both bounding boxes and transcripts of text instances. Current SoA methods commonly produce word-level instances. The proposed pipeline supports off-the-shelf text spotters as a drop-in replacement in the word spotting module.

3.2.2. Place Text Identification

The process aims to eliminate noisy text (e.g., product/price boards, banners, billboards, etc.) and recognizes place text according to the layout (words, lines, or regions).

This is achieved through three sub-steps: bounding fitting, layout analysis, and filtering by occupancy area.

Bounding Fitting: The study [35] empirically demonstrates that a cubic Bezier curve is sufficient to fit different formats of curved scene text. Therefore, the bezier curve is chosen as the common representation, since it can represent varied bounding shapes (e.g., quadrilateral, polygons) produced by different text spotting algorithms, with a fixed number of points. The Bezier curve is a parametric curve, that defines a smooth, continuous curve with a set of discrete control points (refer to Equation (1a)).

$$C(t) = \sum_{i=0}^{n} b_{i,n}(t) P_i, \quad 0 \leq t \leq 1 \tag{1a}$$

$$b_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i}, \quad i = 0, \ldots, n \tag{1b}$$

where, n is the degree , P_i is the ith control point and $b_{i,n}$ is the Bernstein basis polynomials of degree n (refer Equation (1b)).

We employ the bezier ground truth generation procedure performed in [35] to fit the $\{X_i\}_{i=1}^{N}$ set of contour points representing different bounding shapes into a cubic Bezier (n = 3) representation. This is achieved by minimizing the linear least squares fitting error (Equation (2)).

$$S = \sum_{k=0}^{N} \|C_3(t_k) - X_k\|^2 \tag{2}$$

Fitting output bounding shapes into Bezier curve representations, enables the use of off-the-shelf text spotters as a drop-in replacement to the word spotting module. Furthermore, the 8-point cubic Bezier curve representation provides a structured understanding of the text position on the image for layout analysis.

Layout Analysis: As discussed in Section 3.1, the display place name texts exist in different layouts: words, lines, and multiple lines. Therefore, we propose to appropriately concatenate the detected word instances. We use the Bezier curve representation of the bounding shape to determine whether two word instances are in the same line or region. This step allows to recognition of full-place text according to the displayed layout.

Area-based filtering: The line/region level instances are then converted into a convex hull to calculate an area score (Equation (3)). Instances below an area score of 0.1 are discarded and remaining instances are ranked based on the area score to obtain the top N occupying instances as place text candidates. This is based on the observation that place name signs tend to be larger. This step helps eliminate irrelevant text instances, minimize the place text candidates, and improve the results by minimizing false positives.

$$Area_{score} = \frac{Text_{Area}}{Image_{Area}} * 1000 \tag{3}$$

3.2.3. Text Similarity Search

In this stage, previously identified place text candidates are matched against the location-relevant listed place names to obtain top K place recognition results. The location-relevant subset is obtained by filtering places within a radius value (r) from the query location (Equation (4)).

$$\begin{gathered} P_Q = \{P \in P_{DB} \mid distance(Q, P_l) \leq r\} \\ distance(Q, P_l) = R * \sqrt{\left(\Delta\lambda * \cos\left(\frac{\varphi_Q + \varphi_P}{2}\right)\right)^2 + \Delta\varphi^2} \\ \text{Where, } Q = (\lambda_Q, \varphi_Q) \text{ and } P_l = (\lambda_P, \varphi_P) \end{gathered} \tag{4}$$

where P_{DB} is the full place directory, P_l is the place location, and P_Q is the location-relevant subset at query location Q.

As discussed in the requirement definition (Section 3.1), retrieval of the listed place name using the recognized place text is not straightforward. Therefore, listed place names are pre-processed to remove special characters, convert accented characters (e.g., è to e), and remove common repetitive words. Given a nearby place name list for a query, words that occur more than N (=3) times are considered repetitive common words; this removes false positives by eliminating words such as cafe, salon, city name (Harajuku, Omotesando), etc.

For the similarity score calculation between the two strings, we use an iterative scoring function that uses the Levenshtein edit distance [46] as the basis to measure similarity (Equation (5)). In the iterative scoring function, for each display, the place text candidate segment-wise similarity is calculated against each nearby listed place name to obtain the place with top matching segment-wise score. This addresses the challenges caused by partial or imperfect detections, mismatched spacing, and differences between the display name and the listed name (Figure 4).

$$Sim_{score}(s_i, \bar{s_i}) = 1 - \frac{D(s_i, \bar{s_i})}{max(s_i, \bar{s_i})} \tag{5}$$

where $D(:)$ stands for the Levenshtein distance [46], and s_i and $\bar{s_i}$ are the recognized display place text and listed place name, respectively. The Levenshtein distance between two words is defined as the minimum number of single-character edits (insert, delete, and substitute) required to change one word to another.

Additionally, irregular layouts make the place name identification challenging. For instance, some places may have the name in multiple lines, whereas others will have the place name and a tagline in multiple lines (Figure 4e) . Therefore, based on experimental results, we separately consider top N text instances of words, lines, and regions to obtain the overall top K results.

3.3. Hierarchical VPR Architecture

As depicted in Figure 3, the proposed architecture first attempts the VPRText. In case of failure (i.e., no matching place names or no text detections), it attempts place recognition through image retrieval. The top matching similarity score determines VPRText's success; if it is below a given threshold, image retrieval is attempted. The motivation behind this is that there are text-less environments, text can be fully occluded, or the text spotting module may fail to detect/recognize challenging text instances (e.g., complicated font styles, blurry text, oriented views).

Image Retrieval Pipeline: The image retrieval module follows the popular pipeline of retrieving the top matching images from a place image database using global image similarity and optionally re-ranking for result refinement [1,2,4]. Deep learned feature extractors [4–6] can be used to create global image descriptors; this is conducted offline for the database images. Given a query image and positioning information, a global image descriptor is created for the query image and matched against the nearby places' database images to find the top matching images to obtain the top K place recognition results. For feature extraction, we use the DELG [5] model. We do not use a reranker in this study as previous studies indicate that geometric verification-based reranking does not improve place recognition results in urban environments [7,47].

4. Dataset

4.1. Tokyo Place Text

As discussed in Section 2, existing datasets limit the evaluation of the proposed method. Therefore, we introduce a purpose-built dataset to evaluate the SoA text spotting algorithm in reading place names and the proposed VPR architecture under various challenging

conditions. Therefore, we have three subsets of images; text spotting query set, place recognition query set, and places database set.

Place Recognition Query Set: comprises 130 day-time and 50 night-time images (180 images in all). The images are further divided into four subsets: general day—80 images, general night—35 images, perceptually aliased day—50 images, and perceptually aliased night—15 images. These query images contain the place identifier(s) as the ground truth. The set aims to evaluate the performance of proposed methods in accurately recognizing the place.

Places Database Set: comprises 960 images of 240 unique POIs. Each POI contains 1–5 images, annotated with a place signature. Place signature corresponds to a place entry in the place directory. The set serves as the database for image retrieval.

Place Directory: is a database where place id, place name, and georeferencing are stored. For evaluation purposes, we generate a static directory sourced from OpenStreetMaps [48]. It comprises place entries from our experimental areas. Each entry contains the place id, place name, longitude, latitude, and optionally other details. Entries are not limited to places depicted on the query images.

Text Spotting Query Set: comprises 150 day-time and 100 night-time images (250 images in all), with only the visible place name text (transcript) and respective language as the ground truth. This set aims to evaluate the ability of text-spotting algorithms in reading the visible place name on the query image.

All images are collected from Tokyo, Japan, in several rounds. Specifically, they were collected in Harajuku, Omotesando, and Yanaka areas using the mobile devices Sony SO-04J, Apple iPhone X, and 13 Pro Max. The images contain GPS information, including latitude, longitude, and horizontal positioning error in metadata. The dataset includes a variety of challenging instances (Figure 6), including varied font styles, font sizes, arbitrary shapes (including horizontal, curved, and vertical), environment conditions (general, illuminated, and seasonal), and perceptually aliased. Perceptually aliased images are collected from a shopping complex with more than 15 stores (with similar storefronts) facing the street. Night-time images are collected during the Christmas season and therefore, display both illumination and seasonal appearance variances. Even though both English and Japanese texts are available in the environment, the presented query set is limited to English place name texts.

Figure 6. Examples from the TokyoPlaceText evaluation dataset: (**a**) General day-time images; (**b**) Night-time images including illumination and seasonal changes; (**c**) Perceptual aliased day time; (**d**) Perceptually aliased night-time/illuminated.

5. Experiments and Results

This section presents the experimental setup and results. We conduct a series of experiments to evaluate the applicability of text spotting algorithms, detection unit, scoring function and place recognition under varied conditions including day, night, illuminated, seasonal, and perceptually aliased. Experiments are presented under the three subsections; the evaluation of (i) text spotters, (ii) the VPRText pipeline, and (iii) the hierarchical VPR architecture.

5.1. Evaluation of Text Spotters

The key component of the proposed pipeline is the word spotting module. The pipeline allows the use of an off-the-shelf text spotter and it is responsible for the detection and recognition of word instances on the query image. The number of studies have proposed E2E text spotting [35,37] models as well as detection [26,27], and recognition [29,32] models. Yet, limited studies have evaluated the applicability of SoA methods in place recognition tasks. Therefore, first, we evaluate the performance of some of these SoA text spotting, detection, and recognition models in accurately recognizing the display place name texts.

Evaluation Metrics: As our interest lies in the ability to accurately recognize the place name text, E2E text spotting evaluation is performed using Normalized Edit Distance (NED), which is formulated as in Equation (6).

$$N.E.D = 1 - \frac{1}{N}\sum_{i=1}^{N}\frac{D(s_i, \bar{s_i})}{max(s_i, \bar{s_i})} \tag{6}$$

where $D(:)$ stands for Levenshtein distance [46], and s_i and $\bar{s_i}$ are the predicted and ground truth transcription, respectively. Levenshtein distance between two words is defined as the minimum number of single-character edits (insert, delete, substitute) required to change one word to another. Therefore, higher N.E.D. values indicate higher reading accuracy.

Evaluation Dataset: The Text Spotting Evaluation Query Set presented in Section 4.1, comprising 150 day-time and 100 night-time images, are used.

Text Spotting Models: Text spotting can be implemented as a unified E2E module or as a two-step module comprising separate detection and recognition modules. Four openly available SoA E2E text spotters (ABCNet [35], Mango [36], Mask RCNN [45] and Text Perceptron [37]) are evaluated on the day and night conditions and using word, line and region level detections. Results are provided in Table 1. Openly available pre-trained models are used for evaluation. We selected the models fine-tuned on the Total Text dataset [49], as it contains 1555 images with 11,459 text instances including horizontal, multi-oriented, and curved texts. For the evaluation of MaskRCNN Detector, Mango, and Text Perceptron, we utilized the DavarOCR framework [50] and the provided pre-trained models fine-tuned on the Total-Text dataset.

Additionally, two-stage text spotters implemented as a combination of recently introduced detector and recognizer modules are also evaluated. Ten text spotters implemented as a combinations of the detectors—DBNet [27], DRRG [26] and recognizers—ABINet [32], SATRN [31], SAR [30], Robust Scanner [29], NRTR [51] are evaluated. Results are presented in Table 2. For the evaluation, we utilized the MMOCR framework [52] and used the provided pre-trained models. Layout analysis module is also evaluated based on different units: word, line, and region.

Table 1. Evaluation of E2E text spotters.

Condition	Method	N.E.D.		
		Word	**Line**	**Region**
Day	Mango	0.668	0.757	0.746
	ABCNet	0.679	0.778	0.684
	Text Perceptron	0.684	0.782	0.692
	Mask RCNN	0.668	0.688	0.746
Night	Mango	0.684	0.757	0.738
	ABCNet	0.652	0.726	0.689
	Text Perceptron	0.704	0.820	0.759
	Mask RCNN	0.684	0.718	0.738

Table 2. Evaluation of two-step text spotters.

Detector	Recognizer	N.E.D.		
		Word	**Line**	**Region**
DRRG	ABINet	0.633	0.630	0.533
	RobustScanner	0.613	0.605	0.508
	SATRN	0.590	0.589	0.488
	NRTR_1/8-1/4	0.505	0.502	0.423
	SAR	0.622	0.615	0.512
DB_r50	ABINet	0.621	0.645	0.586
	RobustScanner	0.604	0.638	0.573
	SATRN	0.605	0.639	0.571
	NRTR_1/8-1/4	0.540	0.568	0.515
	SAR	0.602	0.637	0.569

5.2. Evaluation of VPRText

Next, we evaluate the performance of our proposed VPRText pipeline. The experiments presented in this section evaluate the place recognition result. The goal is to identify the place from the map/directory using the recognized transcript. As described in Section 3, the pipeline support any off-the-shelf text spotter and output word, line, or region level recognition.

Evaluation Metrics: As for our downstream task, higher recall is more important than precision (as it is not a critical task); we evaluate this using the recall@K metric. We formulate recall@K as in Equation (7c) for the place recognition task.

$$rel(Q)_{@K} = \begin{cases} 1 & \text{if } |R_K \cap R_{gt}| \geq 1 \\ 0 & \text{if } |R_K \cap R_{gt}| = 0 \end{cases} \tag{7a}$$

$$rel(Q)_{@K} = \frac{|R_K \cap R_{gt}|}{min(K, |R_{gt}|)} \tag{7b}$$

$$\mu Recall@K = \frac{1}{N} \sum_{i=1}^{N} rel(Q_i)_{@K} \tag{7c}$$

where $rel(Q)_{@K}$ is the relevance function, which is the availability of ground truth R_{gt} among the retrieved top K results (R_K) for a query image. In addition, $\mu Recall@K$ is relevance averaged over for N query images. Equations (7a) and (7b) refers to the cases of single and multiple POI evaluation, respectively. The study uses the single POI evaluation.

Evaluation Dataset: The Place Recognition Evaluation Query Set presented in Section 4.1 is used. The dataset comprises general, illuminated, seasonal, and perceptually aliased instances.

Baseline: Image retrieval approach is used as the baseline. The image retrieval pipeline is implemented using the DELG [5] global image descriptors with dot product similarity. We use the DELG feature extractor with ResNet101 backbone trained on Google Landmark

Dataset V2 [6]. Image scales given in the original configuration and a 2048 sized global image feature vector are used. The database set described in Section 4.1 is used.

Parameters: VPRText pipeline uses Text Perceptron [37] as the word spotting module, all (line, word, and region) as layout unit, N = 3 for top place text candidate selection, and location filter value(r) of 100 m, and an area threshold of 0.1 is used. Table 3 presents the evaluation of submodules for parameter selection.

Table 3. Evaluation of submodules.

Criteria	Unit	*μRecall@K* (%)			
		K = 1	**K = 3**	**K = 5**	**K = 10**
Word Spotting	Text Perceptron	87.28	88.44	88.44	89.02
	ABCNet	83.24	87.28	87.28	87.28
	MANGO	82.08	86.71	87.86	87.86
	Mask RCNN	81.50	86.71	87.86	87.86
Text Unit	Word	78.95	85.53	85.53	86.84
	Line	82.89	86.84	86.84	86.84
	Region	80.26	86.84	86.84	86.84
	Alll	84.21	90.79	90.79	90.79
Iterative Scoring	with	84.21	90.79	90.79	90.79
	without	39.47	39.47	40.79	40.79
Place Text ID	with (N = 3)	84.21	90.79	90.79	90.79
	without	76.32	82.89	88.16	88.16

5.2.1. Appearance Variance—Day and Night Condition

We evaluate the performance of VPRText under appearance variance using day and night query image sets and compare it with the image retrieval approach. Table 4 presents the results. It can be observed that the VPRText performs significantly better (11% in the day-time and 24% in the night-time) than the image retrieval approach.

Table 4. Comparison between VPRText vs image retrieval approaches in day/night conditions.

Condition	Method	*μRecall@K* (%)			
		K = 1	**K = 3**	**K = 5**	**K = 10**
Day	VPRText	84.13	84.92	84.92	85.71
	Image Retrieval	73.44	75.78	82.03	92.19
Night	VPRText	95.74	97.87	97.87	97.87
	Image Retrieval	71.43	81.63	85.71	93.88

5.2.2. Perceptual Aliasing

We further evaluate the performance of VPRText in a perceptually aliased environment. As described in Section 4.1, our evaluation dataset is divided into four subsets: general day, perceptually aliased day (Day-PA), general night, and perceptually aliased night (Night-PA). Table 5 shows the comparative results for these conditions separately.

It can be observed that VPRText performs well under perceptually aliased environments, while image retrieval struggles. VPRText perform better by 40% and 64% for day and night (perceptually aliased), respectively. VPRText performs better in night/illuminated conditions as well, by more than 5%. Yet, image retrieval has performed better in general day conditions by about 8%.

5.3. Evaluation of Hierarchical Architecture

To perform robustly in text-less environments, we propose the hierarchical architecture combining VPRText and image retrieval (referred to as VPRTextImage). For the VPRText component, we follow the same implementation used in the VPRText evaluation. For the

image retrieval component, we use the DELG [5] image descriptors and dot product similarity score. In cases of no text detections or the top matching similarity score being below a threshold of 0.65, the system falls back to the image retrieval approach. Results are presented in Table 5. It can be observed that VPRTextImage performs the best overall. Combining VPRText with image retrieval has improved the performance under appearance variance, perceptual aliasing as well as general day-time.

Table 5. Comparison between general and perceptually aliased environments.

Condition	Method	*μRecall@K* (%)			
		K = 1	**K = 3**	**K = 5**	**K = 10**
Day-General	Image Retrieval	96.15	100.00	100.00	100.00
	VPRText	88.16	88.16	88.16	89.47
	VPRTextImage	90.79	94.74	94.74	94.74
Day-PA	Image Retrieval	38.00	38.00	54.00	80.00
	VPRText	78.00	80.00	80.00	80.00
	VPRTextImage	80.00	82.00	88.00	92.00
Night-General	Image Retrieval	91.18	94.12	94.12	100.00
	VPRText	96.97	96.97	96.97	96.97
	VPRTextImage	96.97	100.00	100.00	100.00
Night-PA	Image Retrieval	28.57	57.14	71.43	85.71
	VPRText	92.86	100.00	100.00	100.00
	VPRTextImage	92.86	100.00	100.00	100.00
Overall	Image Retrieval	72.88	77.40	83.05	92.66
	VPRText	87.28	88.44	88.44	89.02
	VPRTextImage	89.02	92.49	94.22	95.38

6. Discussion

6.1. Text Spotters

The study evaluates several SoA text spotting methods for the task of place recognition under challenging conditions including varied font styles, layouts, viewpoints, environmental conditions (day, night, illuminated), occlusion, etc. It can be observed that the evaluated E2E text spotters perform significantly better compared to two-step implementations (Table 1 vs. Table 2). In E2E algorithms, the sequential detection and recognition modules share information and are jointly optimized to improve the performance. This provides an example of the easy adaptation of E2E method to a new domain [22].

Among the E2E text spotters Text Perceptron [37], ABCNet [35], Mango [36] and MaskRCNN [45], Text Perceptron performs the best. It is interesting to note that this performance ranking is different from the popular benchmarking datasets such as Total-text [49]. Thus, this emphasizes the need for the evaluation of real-world tasks. Considering the evaluation layout unit, line-level has performed best in most cases (Table 1). This is explainable by the fact that the majority of place name boards are in multiple words in single-line format.

It is observed that these spotters commonly struggle with text instances including complicated font styles (uncommon, cursive), vertical texts, single characters, wide-spaced texts, and oriented views (Figure 7). We encourage researchers to explore techniques robust against these challenging conditions as well as the performance gains that can be achieved in current algorithms through fine-tuning with synthetic data/real-world data comprising the above challenging instances.

Figure 7. Instances where VPRText fails to accurately detect or recognize the place names. (1,2,3)—oriented views, (4,5)—challenging font styles, sizes, and symbols, (6,7)—oriented illuminated views, (8,9)—vertical texts, and (10)—misleading textures.

6.2. VPRText

We propose a pipeline catered towards the VPR through text reading. Since the pipeline supports off-the-shelf text spotters with varied output bounding shapes, it can leverage the latest SoA algorithms without a hassle. This will also allow the pipeline to be used as a benchmark in evaluating text spotters for VPR tasks. Additionally, it can also be adapted to automate the dataset annotation process for VPR tasks by replacing the manual image annotation process.

VPRText pipeline is evaluated under challenging conditions, including illuminated, seasonal, perceptually aliased, cluttered backgrounds, occluded, etc., and compared against the image retrieval approach. Compared to the image retrieval approach [5], the VPRText demonstrates superior performance under challenging perceptually aliased environments, under illumination, seasonal changes and competitive performance under general day conditions (Tables 4 and 5). This could be explained by the fact that visual similarity alone is insufficient when two entities look similar or different instances of the same entity looks different. Figure 8 shows some instances where VPRText performs better to the image retrieval approach. In summary, VPRText performs well overall, whereas image retrieval struggles in challenging conditions. VPRText failure cases indicate that word spotting has failed (Figure 7) in challenging font styles, vertical texts, and some oriented views. Therefore, further improving the word spotting module can lead to enhanced performance.

Most text spotters commonly output at the word level. Yet, when considering the place names in the real world, they occur in different layouts including words, lines, or multi-lines (regions). Therefore, concatenating word level spottings into line or region levels, to obtain the best match, has helped improve the results. Accurate place name prediction is challenging in some instances due to the differences between display name and listed name, partial or inaccurate recognition, inability to separate place name and tagline, common words, repetitive words, special characters, accented characters and spacing. Cleaning listed names by removing symbols, accents, and frequent words, adapting an iterative scoring algorithm and considering different units (words, lines, and regions) have helped overcome these challenges and improved performance by more than 44% (Table 3).

Figure 8. Image retrieval approach vs. VPRText. Examples where VPRText performs better compared to image retrieval in challenging conditions: (**a**,**b**)—perceptually aliased—day, (**c**,**d**)—Perceptually aliased—night, and (**e**,**f**)—general night/illuminated. First column shows the query image, followed by the top 3 image retrieval results, and the last column shows the VPRText output.

VPRText has many advantages over an image retrieval approach. The image retrieval approach's success highly depends on the quality of the database; it requires several images of a place from different viewpoints and under varied conditions [11]. In addition,

image descriptors should be created and stored [3,5]. Therefore, the creation, maintenance, and storage of the database is an extra overhead. In contrast, the text-based approach only requires a simple directory of place names and locations. The coverage is easily expandable since text search queries can easily be made to a third-party service (e.g., search engine, mapping service) to retrieve information, without depending on a single database. Image retrieval is an expensive process, given a query image; it requires the descriptor creation, image similarity measuring against the database, and optionally further re-ranking for result refinement [3–5]. Whereas, the text-based approach requires only the processing of the query image, followed by simple post-processing, which is cost-efficient. Additionally, the image retrieval approach fails to separately identify multiple POIs located in the same building [7]; the text-based approach is capable of handling the challenge.

Text-based methods proposed for visual localization or camera pose estimation tasks use text augmented topological maps and text descriptors depending on the task requirement [14,15,20]. We demonstrate that for place recognition, direct text transcripts can be matched against a place directory to obtain the place identifier.

6.3. Hierarchical System

VPR through text reading and image retrieval has its own strengths; thus, they are complementary. Therefore, we propose a hierarchical VPR architecture that leverages VPRText and image retrieval together with the positioning information.

Approximate positioning information alone is not capable of recognizing the place accurately due to the difference between query and POI location; at a given position, different viewpoints may depict different POIs, and retrieval by location may result in multiple POIs [7]. Even though the image retrieval approach has advanced significantly with the adaptation of deep learned image features [5,6] , it still struggles under extreme appearance variance and perceptually aliasing [2]. Text-based VPR performs well under these challenging conditions. However, it is not applicable to a textless environment and may struggle when the text is fully occluded or fails to detect the text instances. Therefore, combining these three types of information (positioning, texts, and visual), can provide a robust and cost-effective solution (Table 5).

The proposed architecture performs well under the extreme appearance variance, perceptual aliasing as well as text-less environments. Initially, it will process the query image and attempt VPRText and will fall back to image retrieval only if unsuccessful. Therefore the average processing time will be lower compared to the image retrieval approach. When implemented as a real-world application, the system can be improved to identify the failing instances of VPRText and textless POIs to minimize the place image database.

6.4. Limitations

One of the limitations of the text-based method is that text spotters need to be trained to be able to adapt to different languages, whereas image retrieval is language-independent. Yet, this will be a one-time preparatory process, and synthetic data generation methods [53] can be used to easily supplement the required training data. Other limitations include text-less environments, the inability to recognize some text instances, and occlusion of the text instances. Therefore, we propose to combine the VPRText and image retrieval.

In this study, our VPR dataset is limited to a urban environment with textual information. Therefore, we encourage further evaluation under a varying setting. The evaluation of the text spotters were limited to monolingual setting (e.g., English). Multilingual performance is another aspect that need to be further explored. In this study, our main aim was to propose a solution that performs well under challenging conditions. We understand that the latency is an important factor in real-world applications and can be further improved.

7. Conclusions

This study proposes a VPR architecture that fuses positioning, textual and visual information. The pipeline first attempts VPR through text reading and falls back into

image retrieval if unsuccessful. We introduce a new purpose-built dataset and evaluate the proposed pipeline under challenging conditions (i.e., appearance variance, perceptual aliasing, challenging font styles, occlusion). The proposed VPRText method performance is superior under perceptual aliasing and extreme appearance variances. Furthermore, we demonstrate that by combining the VPR through text reading and image retrieval, we can achieve a robust performance under challenging conditions as well as text-less environments. Through empirical evaluation, we demonstrate that scene-text spotters are capable of generalization. However, they can further be improved to accommodate challenging text instances. As latency is essential for real-world applications, we plan to investigate the applicability of mobile-friendly models for text spotting and feature extraction in future work. We believe our proposed architecture can be implemented as an application to support users and explore unfamiliar areas with ease. Such implementations will also benefit cities and business owners by increasing the visibility and accessibility of POIs.

Author Contributions: Conceptualization, S.K. and D.H.; methodology, D.H.; software, D.H.; validation, D.H.; formal analysis, D.H.; investigation, D.H.; resources, S.K.; data curation, D.H.; writing—original draft preparation, D.H.; writing—review and editing, Y.T. and H.Y.; visualization, D.H. and Y.T.; supervision, S.K.; project administration, S.K.; funding acquisition, S.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to thank Reviewers for taking the time and effort necessary to review the manuscript. We sincerely appreciate all valuable comments and suggestions, which helped us to improve the quality of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MDPI	Multidisciplinary Digital Publishing Institute
VPR	Visual Place Recognition
POI	Place of Interest
SoA	State of the Art
GPS	Global Positioning System
E2E	end to end
SLAM	Simultaneous Localization and Mapping

References

1. Masone, C.; Caputo, B. A Survey on Deep Visual Place Recognition. *IEEE Access* **2021**, *9*, 19516–19547. [CrossRef]
2. Garg, S.; Fischer, T.; Milford, M. Where Is Your Place, Visual Place Recognition? In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, Montreal, QC, Canada, 19–27 August 2021; pp. 4416–4425. [CrossRef]
3. Arandjelović, R.; Gronat, P.; Torii, A.; Pajdla, T.; Sivic, J. NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 1437–1451. [CrossRef] [PubMed]
4. Noh, H.; Araujo, A.; Sim, J.; Weyand, T.; Han, B. Large-Scale Image Retrieval with Attentive Deep Local Features. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 3476–3485. [CrossRef]
5. Cao, B.; Araujo, A.; Sim, J. Unifying Deep Local and Global Features for Image Search. In Proceedings of the Computer Vision–ECCV 2020, Glasgow, UK, 23–28 August 2020; pp. 726–743.
6. Weyand, T.; Araujo, A.; Cao, B.; Sim, J. Google Landmarks Dataset v2—A Large-Scale Benchmark for Instance-Level Recognition and Retrieval. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June, 2020; pp. 2572–2581. [CrossRef]
7. Hettiarachchi, D.; Kamijo, S. Visual and Positioning Information Fusion Towards Urban Place Recognition. *SN Comput. Sci.* **2023**, *4*, 44. [CrossRef]

8. Garg, S.; Suenderhauf, N.; Milford, M. LoST? Appearance-Invariant Place Recognition for Opposite Viewpoints using Visual Semantics. In Proceedings of the Robotics: Science and Systems XIV, Pittsburgh, PA, USA, 16–30 June 2018.
9. Khaliq, A.; Ehsan, S.; Chen, Z.; Milford, M.; McDonald-Maier, K. A Holistic Visual Place Recognition Approach Using Lightweight CNNs for Significant ViewPoint and Appearance Changes. *IEEE Trans. Robot.* **2020**, *36*, 561–569. [CrossRef]
10. Doan, D.; Latif, Y.; Chin, T.J.; Liu, Y.; Do, T.T.; Reid, I. Scalable Place Recognition Under Appearance Change for Autonomous Driving. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9318–9327. [CrossRef]
11. Churchill, W.; Newman, P. Experience-based navigation for long-term localisation. *Int. J. Robot. Res.* **2013**, *32*, 1645–1661. [CrossRef]
12. Porav, H.; Maddern, W.; Newman, P. Adversarial Training for Adverse Conditions: Robust Metric Localisation Using Appearance Transfer. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 1011–1018. [CrossRef]
13. Anoosheh, A.; Sattler, T.; Timofte, R.; Pollefeys, M.; Gool, L.V. Night-to-Day Image Translation for Retrieval-based Localization. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 5958–5964. [CrossRef]
14. Hong, Z.; Petillot, Y.; Lane, D.; Miao, Y.; Wang, S. TextPlace: Visual Place Recognition and Topological Localization Through Reading Scene Texts. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 2861–2870. [CrossRef]
15. Li, B.; Zou, D.; Sartori, D.; Pei, L.; Yu, W. TextSLAM: Visual SLAM with Planar Text Features. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 2102–2108. [CrossRef]
16. Ge, G.; Zhang, Y.; Wang, W.; Jiang, Q.; Hu, L.; Wang, Y. Text-MCL: Autonomous mobile robot localization in similar environment using text-level semantic information. *Machines* **2022**, *10*, 169. [CrossRef]
17. Teichmann, M.; Araujo, A.; Zhu, M.; Sim, J. Detect-To-Retrieve: Efficient Regional Aggregation for Image Search. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 5104–5113. [CrossRef]
18. Torii, A.; Taira, H.; Sivic, J.; Pollefeys, M.; Okutomi, M.; Pajdla, T.; Sattler, T. Are Large-Scale 3D Models Really Necessary for Accurate Visual Localization? *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 814–829. [CrossRef] [PubMed]
19. Radwan, N.; Tipaldi, G.D.; Spinello, L.; Burgard, W. Do you see the bakery? Leveraging geo-referenced texts for global localization in public maps. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 4837–4842. [CrossRef]
20. Yu, J.; Su, J. Visual Place Recognition via Semantic and Geometric Descriptor for Automated Valet Parking. In Proceedings of the 2021 IEEE International Conference on Robotics and Biomimetics (ROBIO), Sanya, China, 6–9 December 2021; pp. 1142–1147. [CrossRef]
21. Long, S.; He, X.; Yao, C. Scene text detection and recognition: The deep learning era. *Int. J. Comput. Vis.* **2021**, *129*, 161–184. [CrossRef]
22. Chen, X.; Jin, L.; Zhu, Y.; Luo, C.; Wang, T. Text recognition in the wild: A survey. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–35. [CrossRef]
23. Tian, S.; Pan, Y.; Huang, C.; Lu, S.; Yu, K.; Tan, C.L. Text Flow: A Unified Text Detection System in Natural Scene Images. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 4651–4659. [CrossRef]
24. Zhou, X.; Yao, C.; Wen, H.; Wang, Y.; Zhou, S.; He, W.; Liang, J. East: An efficient and accurate scene text detector. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5551–5560.
25. Ye, J.; Chen, Z.; Liu, J.; Du, B. TextFuseNet: Scene Text Detection with Richer Fused Features. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, Yokohama, Japan, 7–15 January 2021; pp. 516–522.
26. Zhang, S.X.; Zhu, X.; Hou, J.B.; Liu, C.; Yang, C.; Wang, H.; Yin, X.C. Deep relational reasoning graph network for arbitrary shape text detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9699–9708.
27. Liao, M.; Wan, Z.; Yao, C.; Chen, K.; Bai, X. Real-Time Scene Text Detection with Differentiable Binarization. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 11474–11481.
28. Shi, B.; Bai, X.; Yao, C. An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2298–2304. [CrossRef] [PubMed]
29. Yue, X.; Kuang, Z.; Lin, C.; Sun, H.; Zhang, W. RobustScanner: Dynamically Enhancing Positional Clues for Robust Text Recognition. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020.
30. Li, H.; Wang, P.; Shen, C.; Zhang, G. Show, attend and read: A simple and strong baseline for irregular text recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 8610–8617.
31. Lee, J.; Park, S.; Baek, J.; Oh, S.J.; Kim, S.; Lee, H. On Recognizing Texts of Arbitrary Shapes with 2D Self-Attention. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 2326–2335. [CrossRef]

32. Fang, S.; Xie, H.; Wang, Y.; Mao, Z.; Zhang, Y. Read Like Humans: Autonomous, Bidirectional and Iterative Language Modeling for Scene Text Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 7098–7107.
33. Bookstein, F. Principal warps: thin-plate splines and the decomposition of deformations. *IEEE Trans. Pattern Anal. Mach. Intell.* **1989**, *11*, 567–585. [CrossRef]
34. Jaderberg, M.; Simonyan, K.; Zisserman, A. Spatial transformer networks. In Proceedings of the Advances in Neural Information Processing Systems 28 (NIPS 2015), Montreal, QC, Canada, 7–12 December 2015.
35. Liu, Y.; Shen, C.; Jin, L.; He, T.; Chen, P.; Liu, C.; Chen, H. ABCNet v2: Adaptive bezier-curve network for real-time end-to-end text spotting. *arXiv* **2021**, arXiv:2105.03620.
36. Qiao, L.; Chen, Y.; Cheng, Z.; Xu, Y.; Niu, Y.; Pu, S.; Wu, F. Mango: A mask attention guided one-stage scene text spotter. *arXiv* **2020**, arXiv:2012.04350.
37. Qiao, L.; Tang, S.; Cheng, Z.; Xu, Y.; Niu, Y.; Pu, S.; Wu, F. Text Perceptron: Towards End-to-End Arbitrary-Shaped Text Spotting. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI), New York, NY, USA, 7–12 February 2020; pp. 11899–11907.
38. Chng, C.K.; Liu, Y.; Sun, Y.; Ng, C.C.; Luo, C.; Ni, Z.; Fang, C.; Zhang, S.; Han, J.; Ding, E.; et al. ICDAR2019 Robust Reading Challenge on Arbitrary-Shaped Text—RRC-ArT. In Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 20–25 September 2019; pp. 1571–1576. [CrossRef]
39. Nayef, N.; Patel, Y.; Busta, M.; Chowdhury, P.N.; Karatzas, D.; Khlif, W.; Matas, J.; Pal, U.; Burie, J.C.; Liu, C.L.; et al. ICDAR2019 Robust Reading Challenge on Multi-lingual Scene Text Detection and Recognition—RRC-MLT-2019. In Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 20–25 September 2019; pp. 1582–1587. doi: 10.1109/ICDAR.2019.00254. [CrossRef]
40. Sun, Y.; Ni, Z.; Chng, C.K.; Liu, Y.; Luo, C.; Ng, C.C.; Han, J.; Ding, E.; Liu, J.; Karatzas, D.; et al. ICDAR 2019 Competition on Large-Scale Street View Text with Partial Labeling—RRC-LSVT. In Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 20–25 September 2019; pp. 1557–1562. [CrossRef]
41. Zhang, C.; Ding, W.; Peng, G.; Fu, F.; Wang, W. Street View Text Recognition With Deep Learning for Urban Scene Understanding in Intelligent Transportation Systems. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 4727–4743. [CrossRef]
42. Zhang, X.; Wang, L.; Su, Y. Visual place recognition: A survey from deep learning perspective. *Pattern Recognit.* **2021**, *113*, 107760. [CrossRef]
43. Torii, A.; Arandjelovic, R.; Sivic, J.; Okutomi, M.; Pajdla, T. 24/7 place recognition by view synthesis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1808–1817.
44. Sun, Y.; Liu, J.; Liu, W.; Han, J.; Ding, E.; Liu, J. Chinese street view text: Large-scale chinese text reading with partially supervised learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9086–9095.
45. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
46. Levenshtein, V.I.. Binary codes capable of correcting deletions, insertions, and reversals. *Sov. Phys. Dokl.* **1966**, *10*, 707–710.
47. Hettiarachchi, D.; Kamijo, S. Visual and Location Information Fusion for Hierarchical Place Recognition. In Proceedings of the 2022 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 7–9 January 2022; pp. 1–6. [CrossRef]
48. Haklay, M.; Weber, P. OpenStreetMap: User-Generated Street Maps. *IEEE Pervasive Comput.* **2008**, *7*, 12–18. [CrossRef]
49. Ch'ng, C.K.; Chan, C.S. Total-text: A comprehensive dataset for scene text detection and recognition. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 1, pp. 935–942.
50. Qiao, L.; Jiang, H.; Chen, Y.; Li, C.; Li, P.; Li, Z.; Zou, B.; Guo, D.; Xu, Y.; Xu, Y.; et al. DavarOCR: A Toolbox for OCR and Multi-Modal Document Understanding. In Proceedings of the 30th ACM International Conference on Multimedia, Lisboa, Portugal, 10–14 October 2022; pp. 7355–7358 doi: 10.1145/3503161.3548547. [CrossRef]
51. Sheng, F.; Chen, Z.; Xu, B. NRTR: A no-recurrence sequence-to-sequence model for scene text recognition. In Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 20–25 September 2019; pp. 781–786.
52. Kuang, Z.; Sun, H.; Li, Z.; Yue, X.; Lin, T.H.; Chen, J.; Wei, H.; Zhu, Y.; Gao, T.; Zhang, W.; et al. MMOCR: A Comprehensive Toolbox for Text Detection, Recognition and Understanding. *arXiv* **2021**, arXiv:2108.06543.
53. Gupta, A.; Vedaldi, A.; Zisserman, A. Synthetic data for text localisation in natural images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2315–2324.