13th IROS Workshop on Planning, Perception, Navigation for Intelligent Vehicle

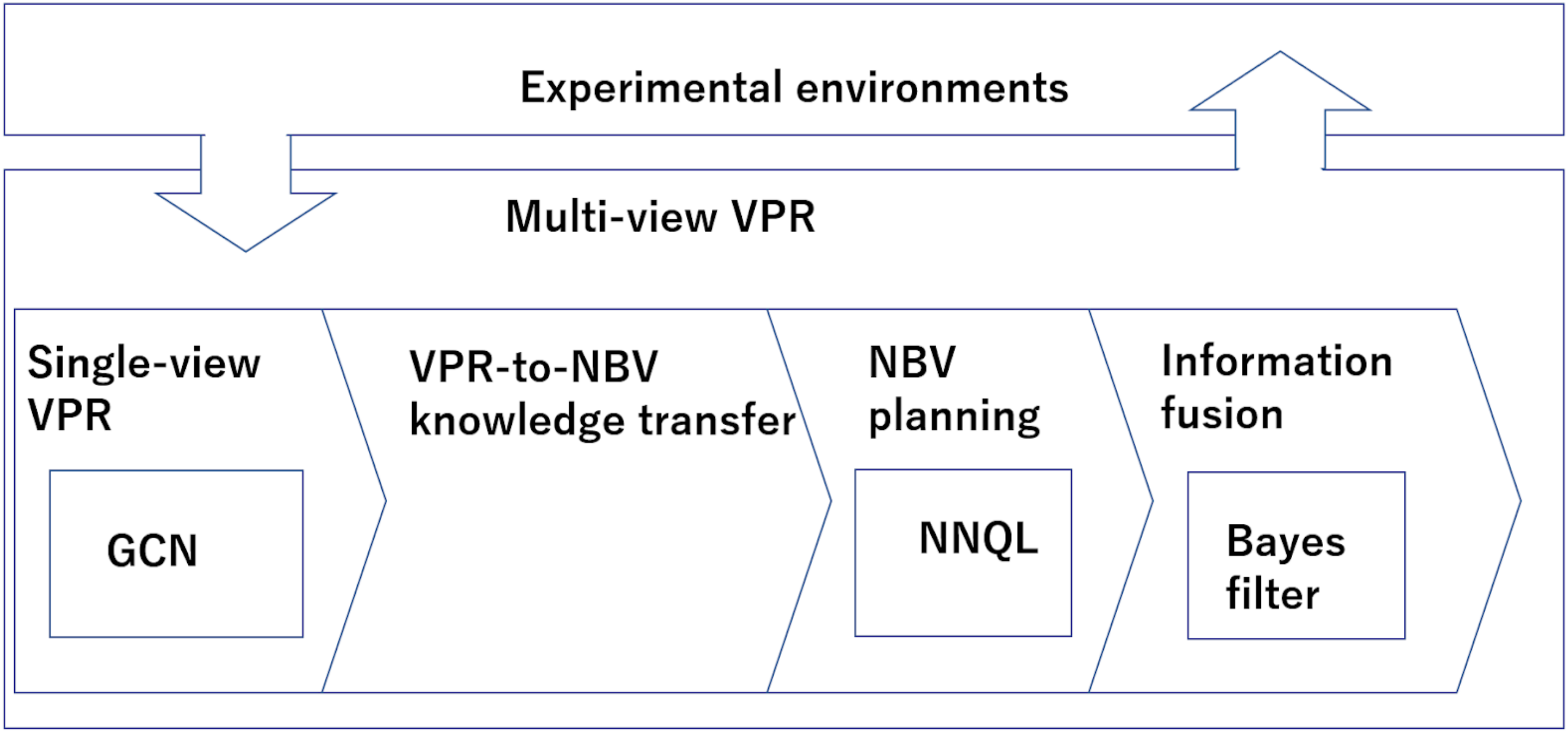
Full Day Workshop, October 23th, 2022, Kyoto, Japan

Highly Compressive Visual Self-localization Using Sequential Semantic Scene Graph and Graph Convolutional Neural Network

Yoshida Mitsuki, Yamamoto Ryogo, Tanaka Kanji

University of Fukui, Japan

We are interested in the problem of robot self-localization or visual place recognition (VPR) using graph convolutional neural networks (GCN). We are motivated by the fact that GCN can directly learn and recognize structured data without relying on non-scalable structured pattern recognition such as pairwise comparison of query and reference scenes, without relying on vector space embeddings.



In our previous ICRA2021 paper, we developed a method to train a GCN visual place classifier by transferring knowledge from a self-localization system using NetVLAD.

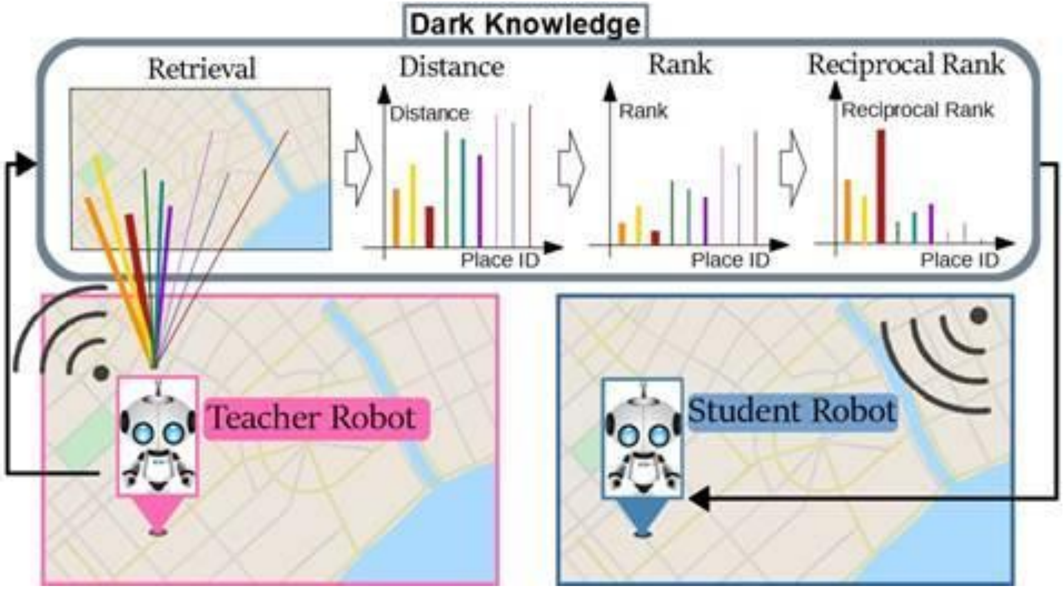


Fig. 1. We propose the use of the reciprocal-rank vector as the dark knowledge to be transferred from a self-localization model (i.e., teacher) to a graph convolutional self-localization network (i.e., student), for improving the self-localization performance.

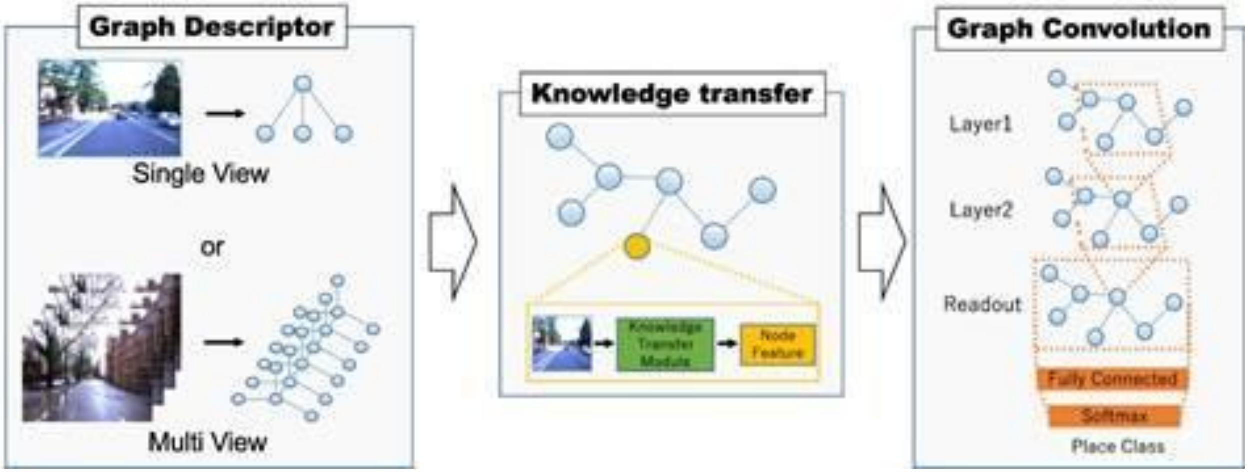
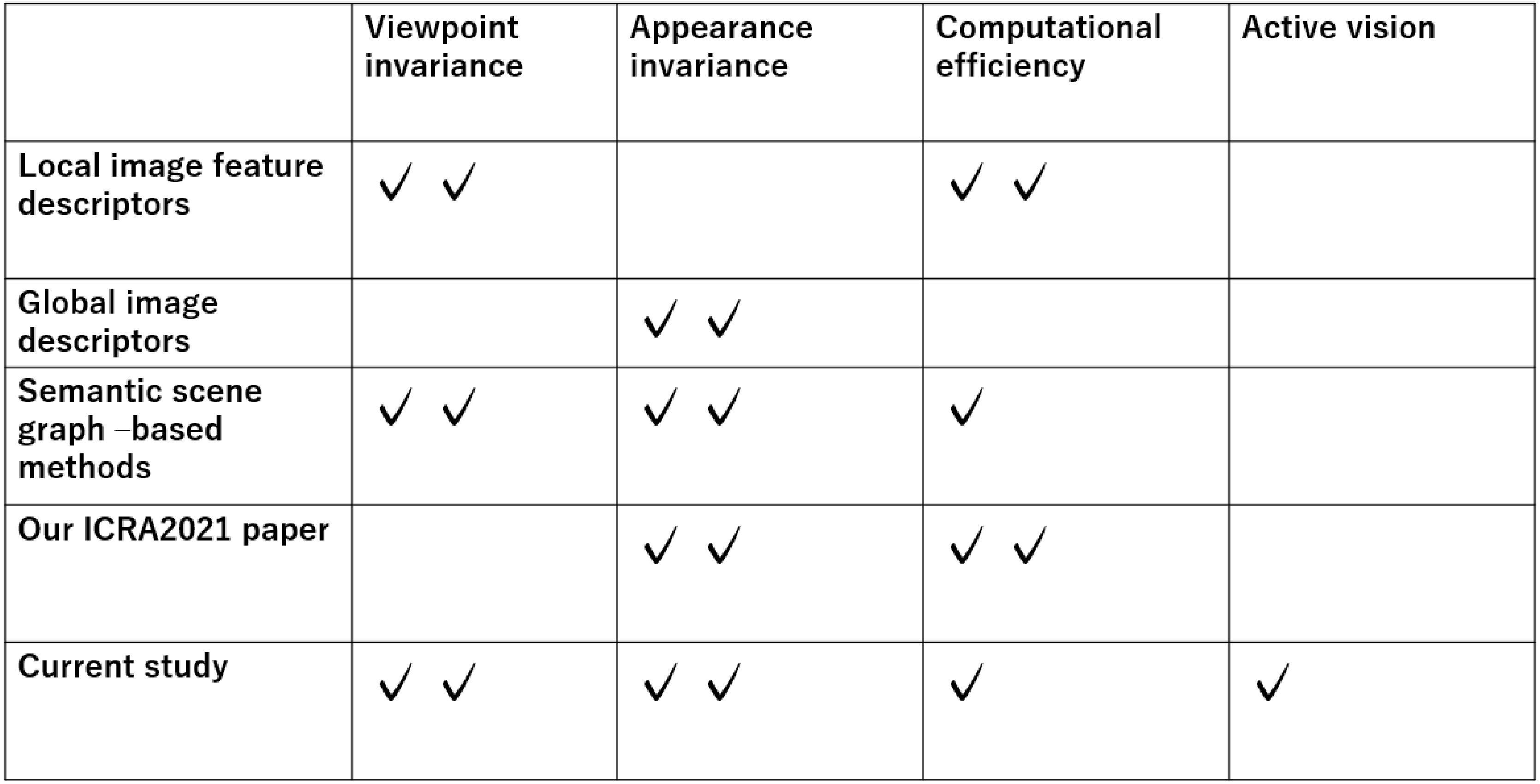


Fig. 2. System architecture.

In the previous work, we assumed a naive self-localization scenario where the same predefined path was traveled for both training and testing. In such a case, VPR can be reduced to a near duplicate image retrieval problem.

In the current study, we aim for the general scenario where the test trajectory can differ significantly from or only partially overlaps the training trajectory. Examples of such scenarios are applications such as patrol robots and mail delivery robots.

At this time, the major source of difficulty is the change in viewpoint. In general, global image descriptors are sensitive to viewpoint changes, and local image descriptors are sensitive to illumination changes.



	Viewpoint invariance	Appearance invariance	Computational efficiency	Active vision
Local image feature descriptors	✓ ✓		✓ ✓	
Global image descriptors		✓ ✓		
Semantic scene graph –based methods	✓ ✓	✓ ✓	✓	
Our ICRA2021 paper		✓ ✓	✓ ✓	
Current study	✓ ✓	✓ ✓	✓	✓

In scene graph-based VPR, semantic scene graphs have proven useful in such scenarios.

A semantic scene graph segments an input scene image into region nodes, where each graph node corresponds to a region of the image, which is described by a semantic label. For example, the X-View method uses an attributed graph that generates region nodes from an image by semantic segmentation and connects adjacent regions with edges. In general, semantic scene graphs inherit robustness and distinctiveness from semantic attributes and scene graphs.

A major limitation of existing semantic attribute graphs is their computational complexity.

A naive way to reduce computational complexity is graph embedding. However, although the matching speed is improved, graph embedding is a complex and unstable computational process.

The GCN used in this research can learn efficiently a lightweight model directly from graph data.

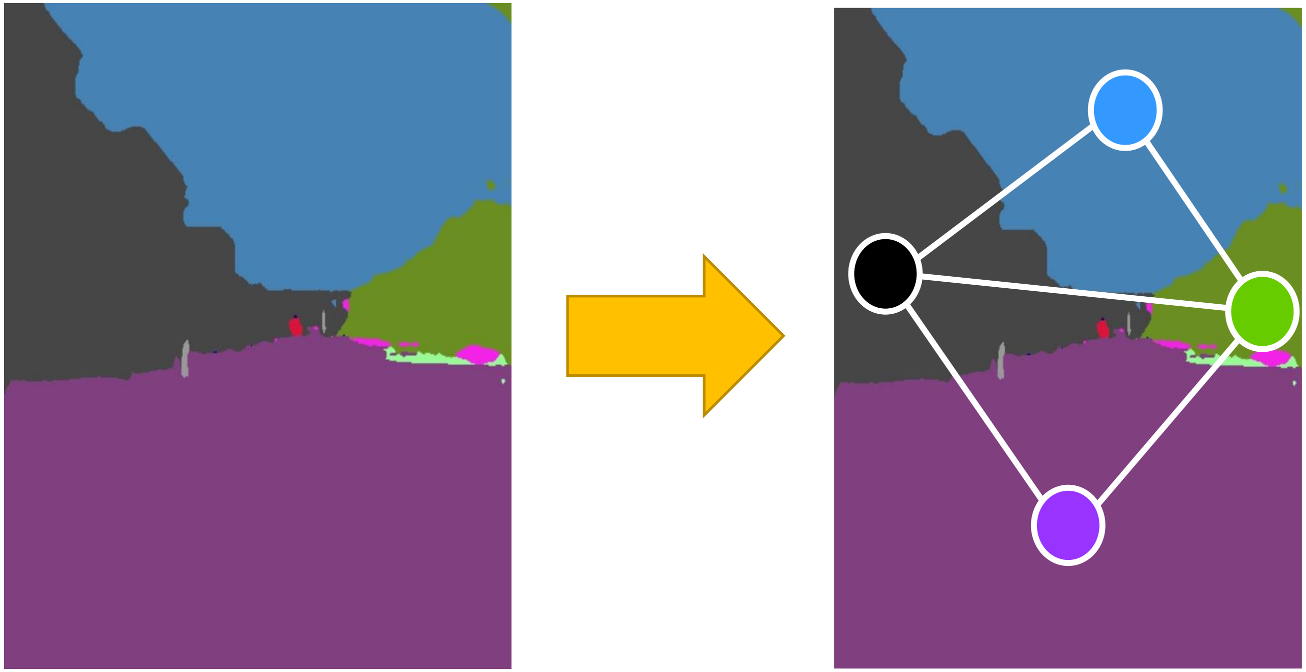
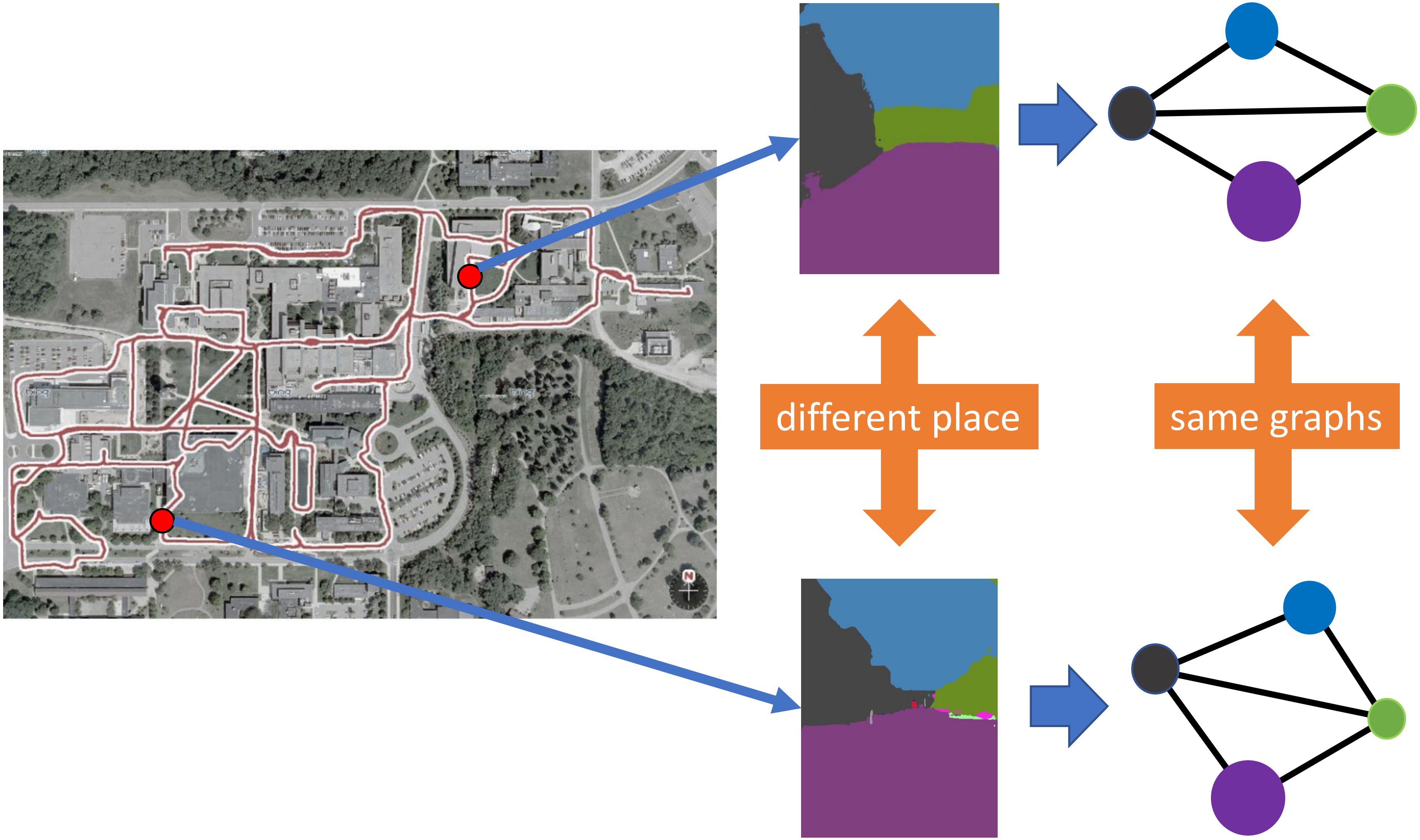


Fig. 1. semantic scene graph by X-View method

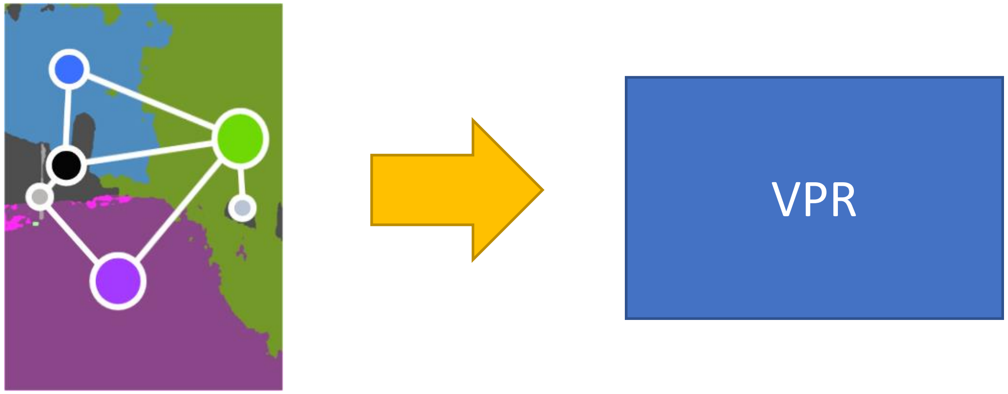
However, semantic scene graphs are limited in terms of discriminatively.
Semantic labels only have a few hundred possible values.
This is not always discriminative in large scale environments.



different place

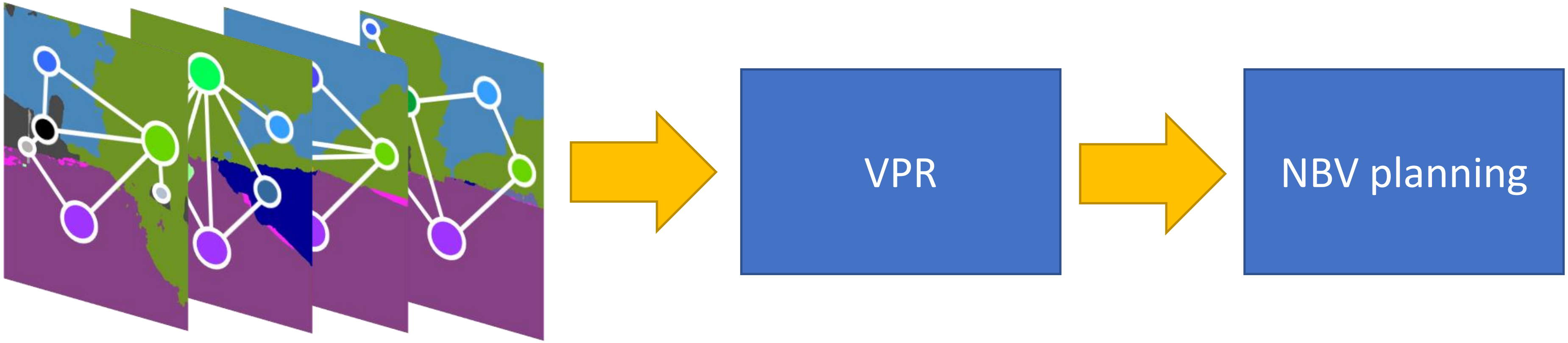
same graphs

To overcome the inherent ill-posedness of single-view VPR, it is extended to multi-view VPR in our approach. Especially, active vision or next-best-view (NBV) approach is considered because the VPR performance is highly dependent on the view sequence. The second contribution of this research is to deal with such NBV planning problem.



VPR

single-view VPR



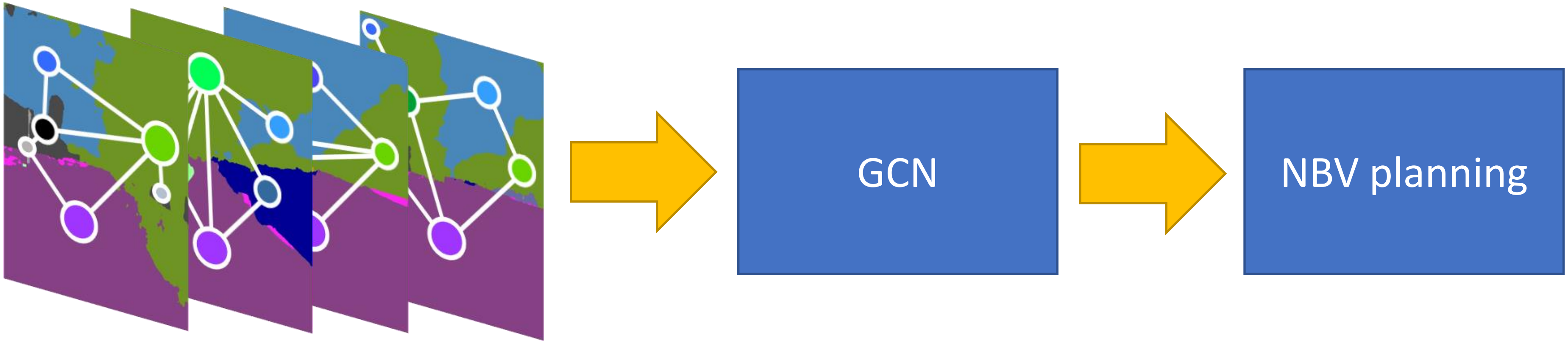
VPR

NBV planning

multi-view VPR

The contribution of our approach can be summarized as follows.

- Training of a GCN classifier on semantic scene graph and its application to robot self-localization
- Proposal of NBV planning to maximize localization performance
- Performance verification by experiment



GCN

NBV planning

This presentation is organized as follows.

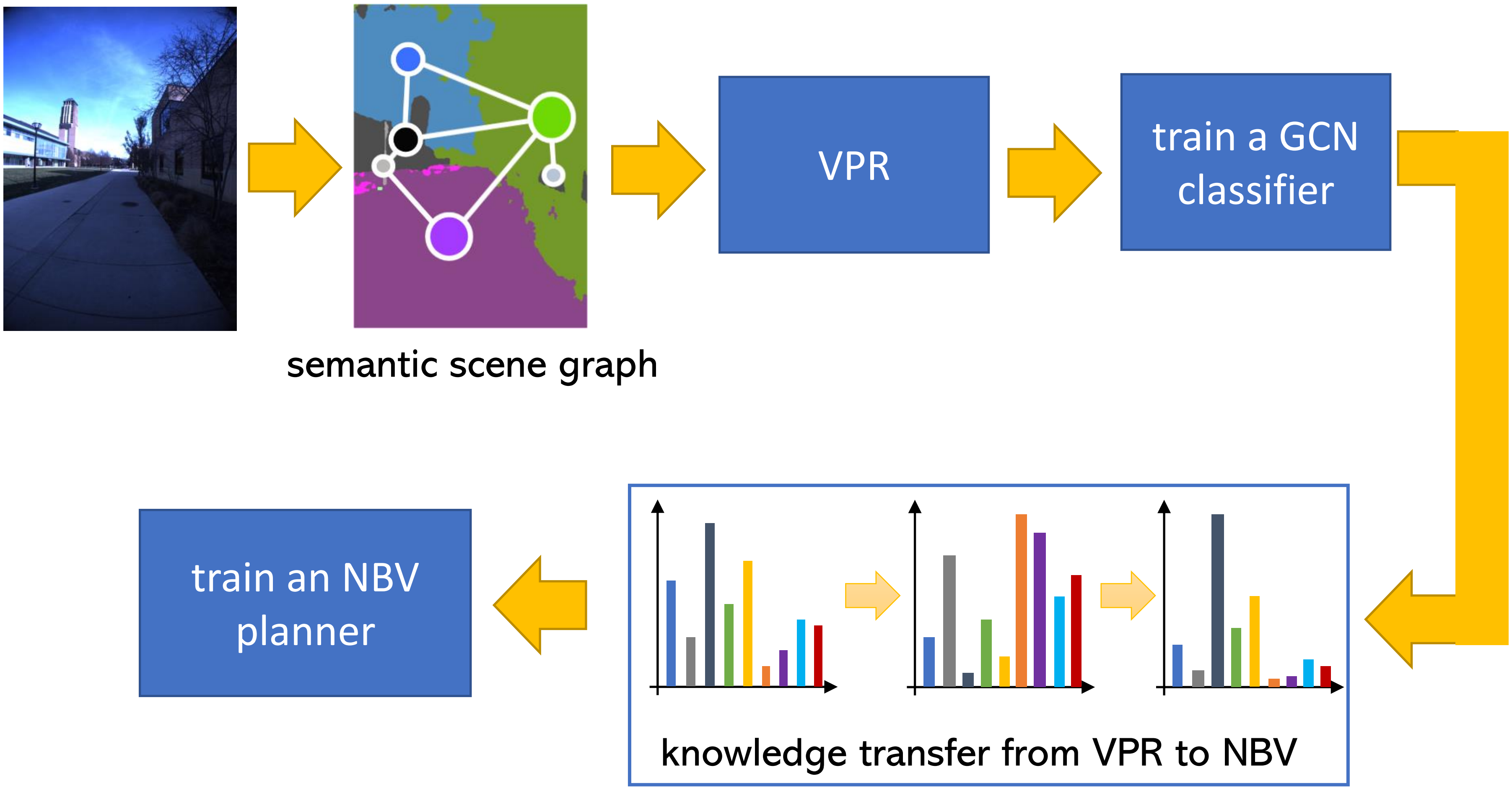
First, we present the semantic scene graph representation used in this study.

Next, we formulate the VPR problem using the scene graph representation.

Next, we show how to train a GCN classifier using the scene graph representation.

Next, we show how to train an NBV planner using a VPR-to-NBV knowledge transfer.

Finally, we present experiments and ablation studies of the proposed method.



semantic scene graph

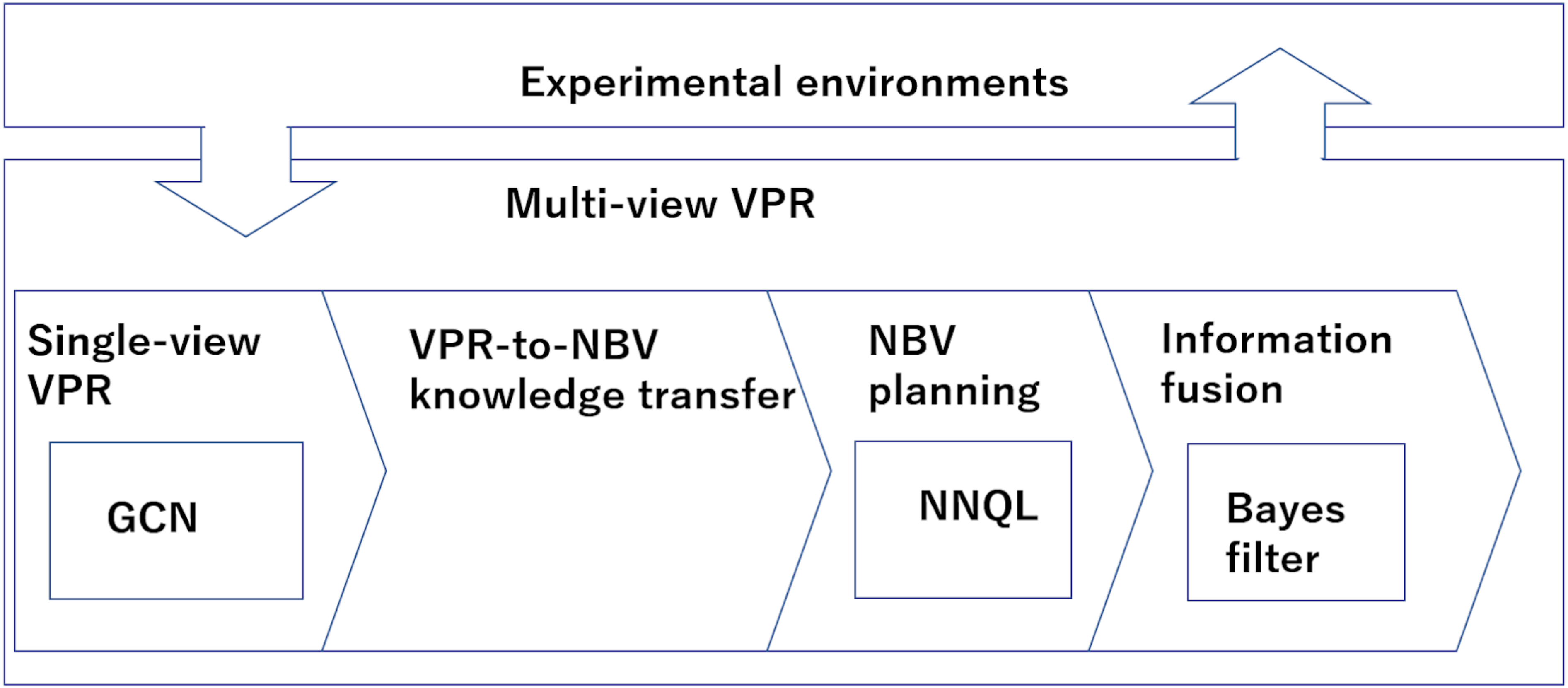
VPR

train a GCN classifier

train an NBV planner

knowledge transfer from VPR to NBV

First, let's talk about the scene graph representation.



The scene graph construction method used here is based on bottom-up semantic segmentation.

① The scene is segmented using the semantic segmentation proposed in Reference 28.

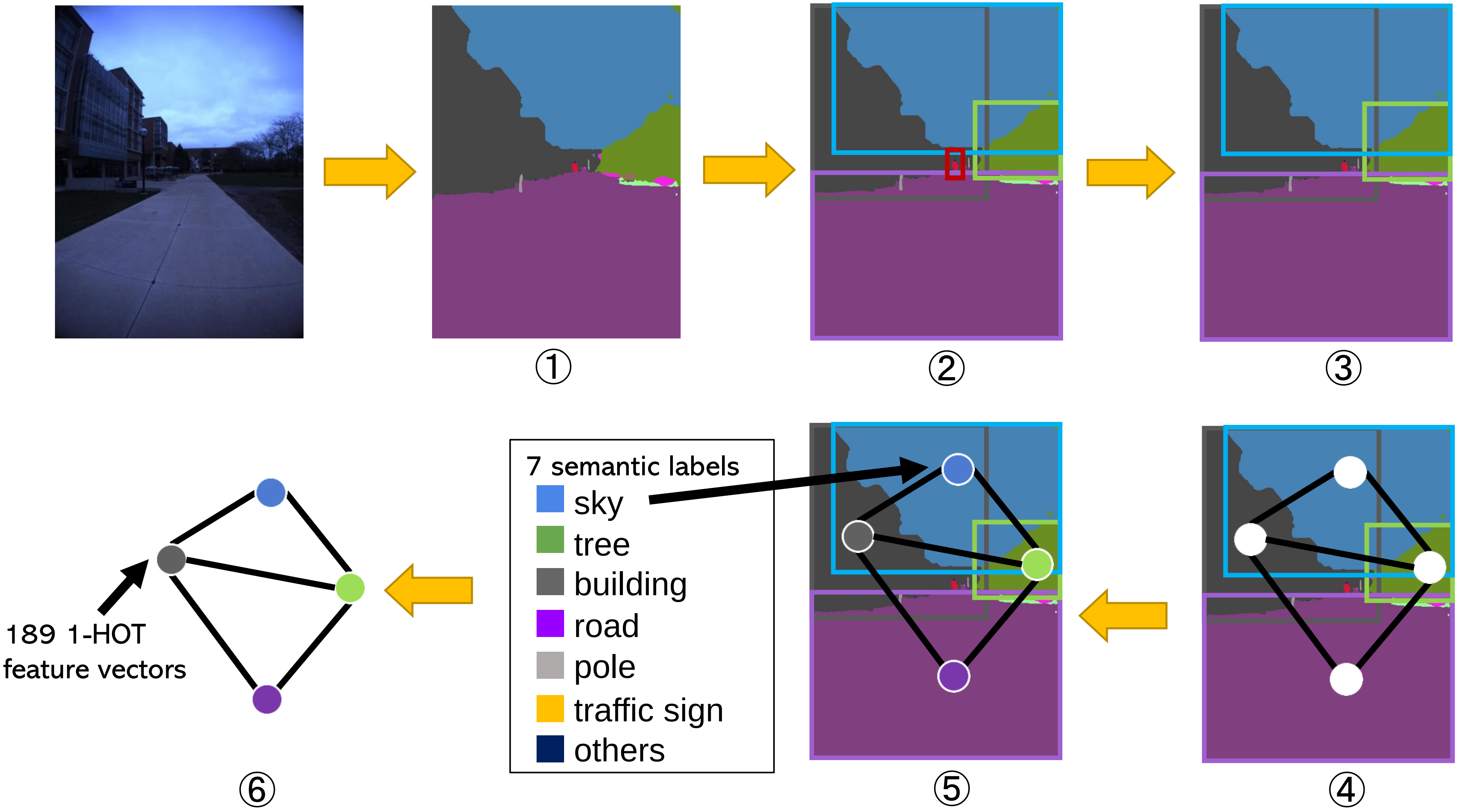
② Then, each region is approximated with a bounding box.

③ Then, bounding boxes with too small areas are considered non-distinctive and removed.

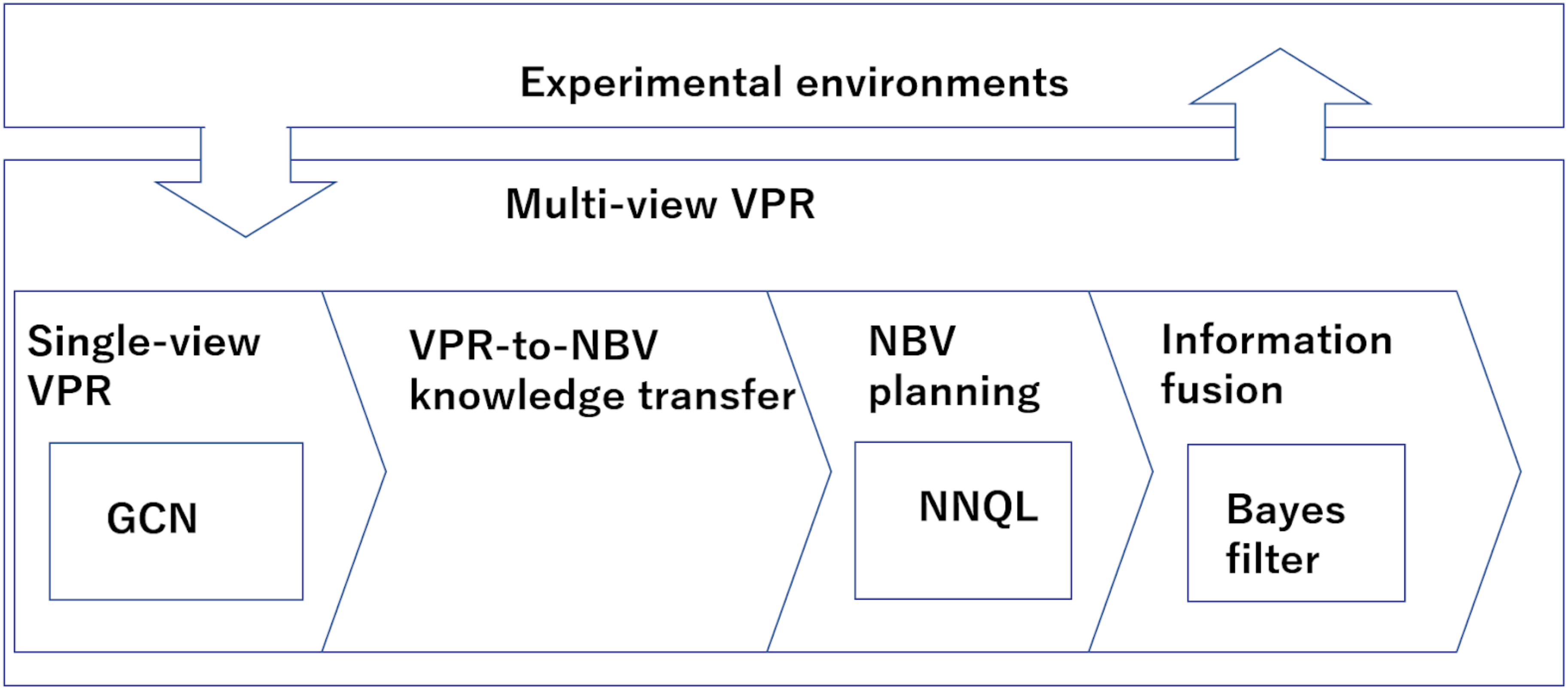
④ Then, regions corresponding to adjacent bounding boxes are connected with graph edges.

⑤ Then, each region is labeled using a dictionary of 7 different semantic labels.

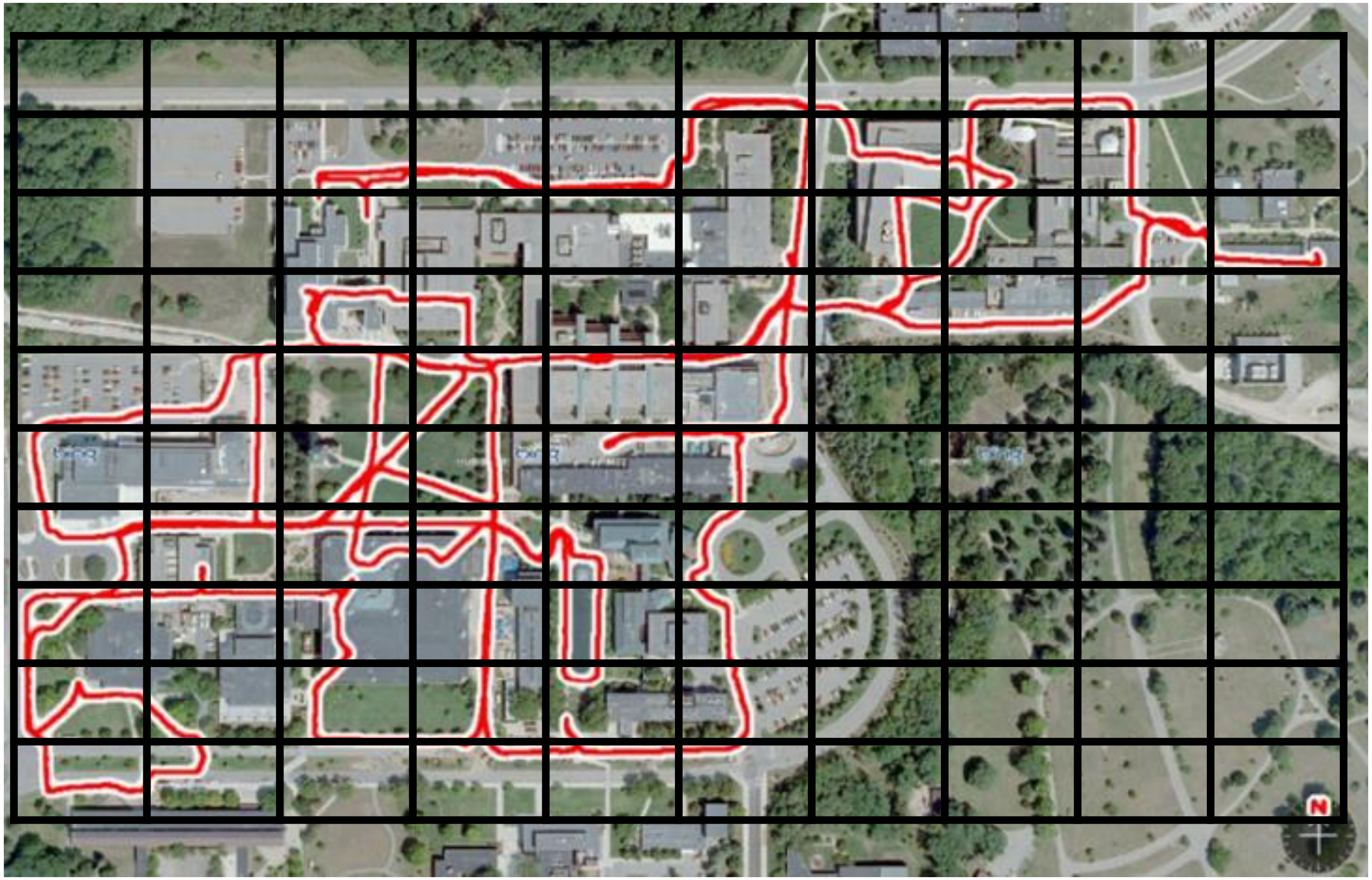
⑥ Then, each region feature is interpreted as 189 1-HOT feature vectors per region.



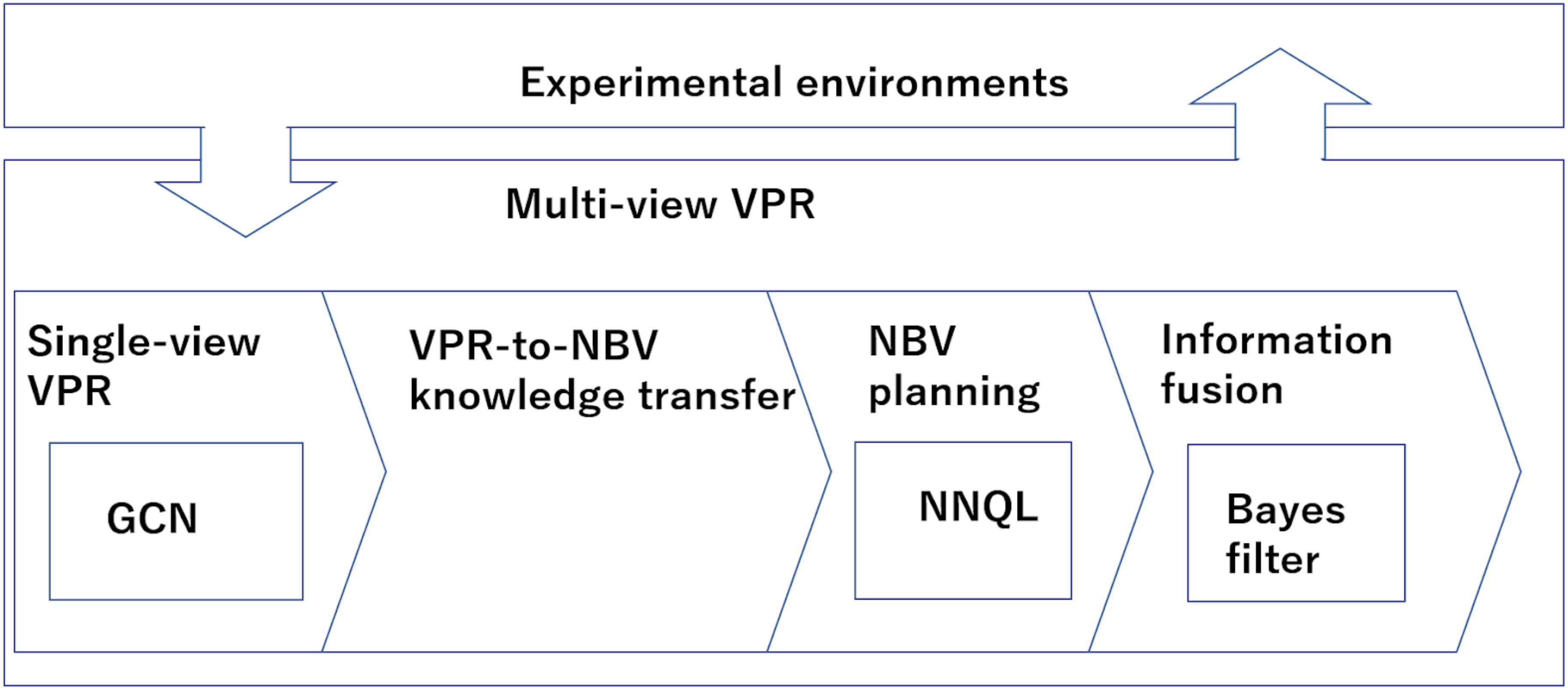
Next, I will explain the VPR problem.



In this study, visual place recognition is formulated as an image classification problem, classifying a visual image into one of a set of predefined place classes. For the place definition, a standard grid-based place partitioning is employed. Specifically, a 10 by 10 grid divides the robot's workspace into 100 regions.



Next, I will explain the training process of the visual place classifier.



As mentioned, a GCN is employed as the scene graph classifier.

In GCN, the graph convolution operation takes a node in the graph and processes it in the following manner.

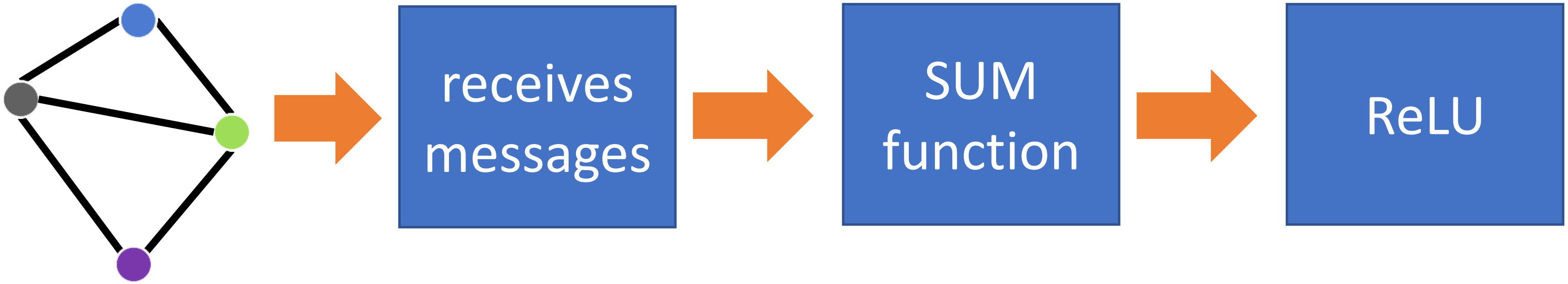
First, it receives messages from nodes connected by the edge.

The collected messages are then summed via the SUM function.

The result is passed through a single-layer fully connected neural network followed by a nonlinear transformation for conversion into a new feature vector.

In this study, the rectified linear unit (ReLU) operation was used as the nonlinear transformation.

Specifically, we used the deep graph library on the Pytorch backend.

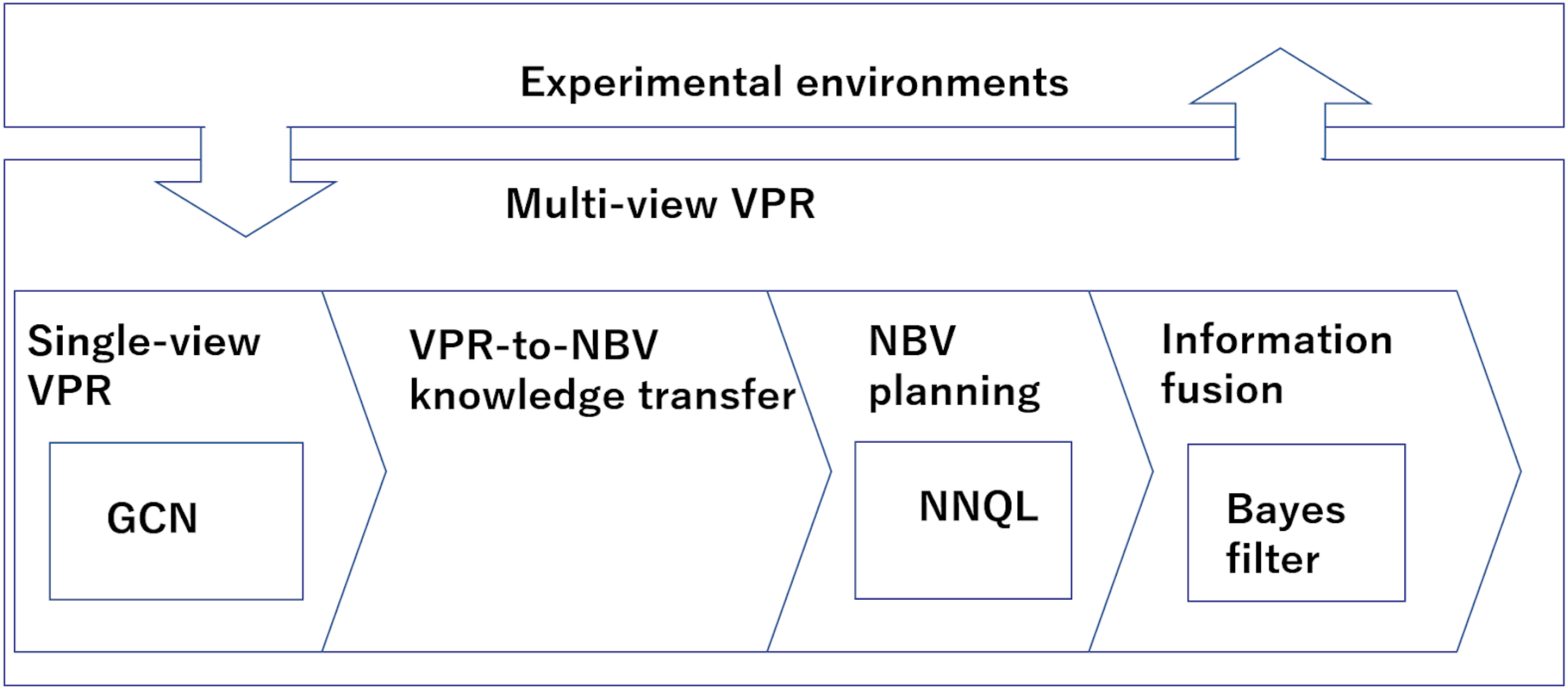


receives messages

SUM function

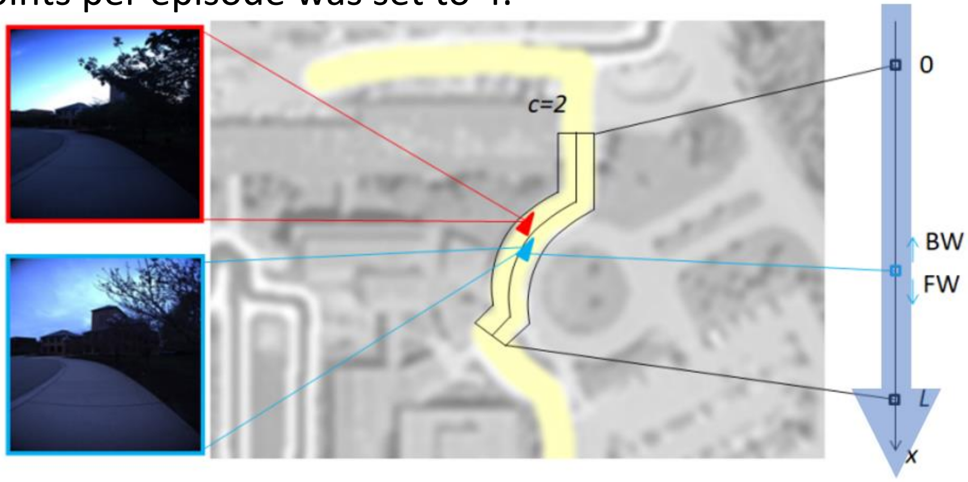
ReLU

Next, I will explain the NBV problem.



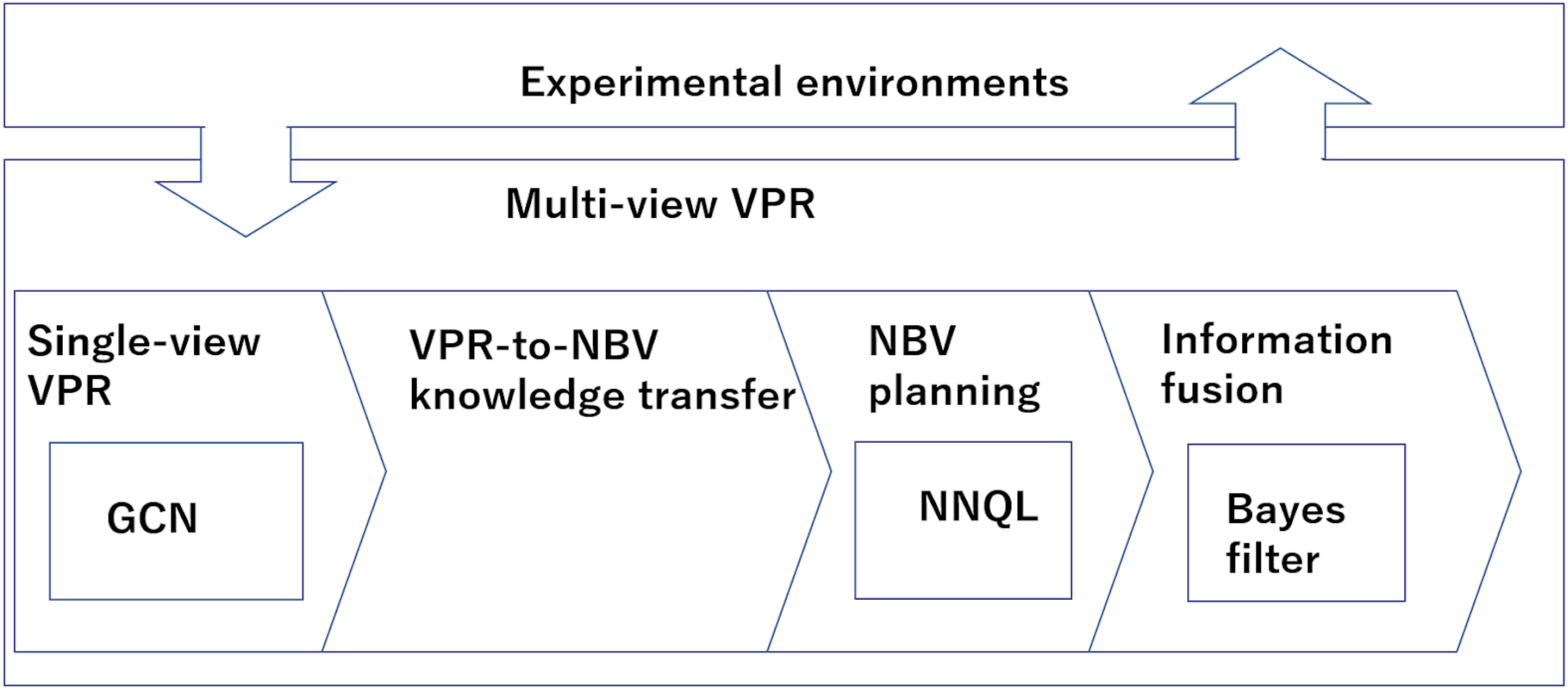
The NBV aims to plan the next best view (NBV) to maximize VPR performance by considering possible future viewpoint trajectories.
Specifically, for example, a robot would like to find landmarks for reliable localization.
Such landmarks may be sparsely distributed in the environment.
Therefore, it is important for the robot to learn to move with a high possibility of finding landmarks.
Specifically, we use the sequential semantic scene graphs (S3G) acquired by the survey robot in the training environment or training domain as training data.
Then, the problem of active VPR is formulated as the learning task of reinforcement learning.

For the sake of simplicity, a 1-dimensional NBV planning problem is considered with a predefined route, as shown in the figure.
This bird's-eye view visualizes the robot's route and action candidates.
The action set is a set of size 10 consisting of forward move candidates from 1 m to 10 m .
We used a fast NNQL algorithm for training reinforcement learning.
A set of 10000 episodes are randomly sampled and used to train NNQL.
The number of viewpoints per episode was set to 4.



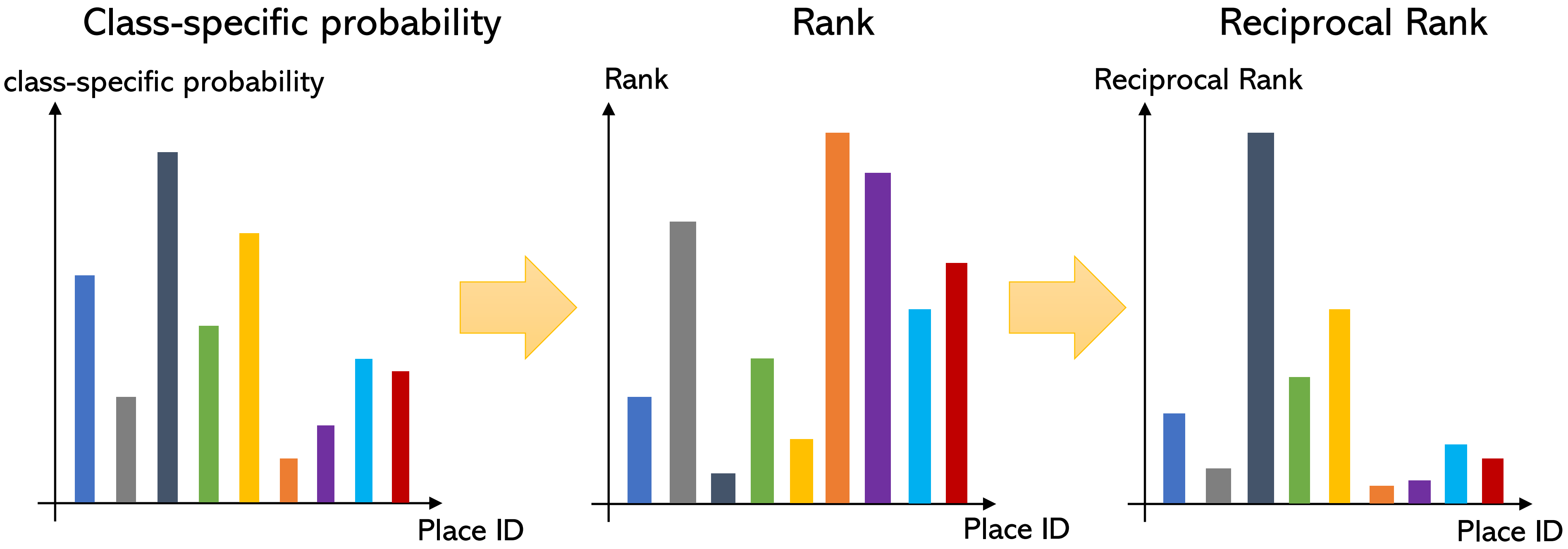
viewpoint
Spatial resolution: 1 [m]

Next, we will introduce a VPR-to-NBV knowledge transfer scheme, which was developed in our previous study.



As already mentioned, a GCN employed as a VPR classifier takes a semantic scene graph as input and outputs a class-specific probability map with the same dimensionality as the size of the place class set. Therefore, GCN can be viewed as an embedding method for semantic scene graphs.

Additionally, we use the reciprocal rank feature developed in our previous work to improve accuracy of this embedding. The procedure for computing the reciprocal rank feature vector from the class-specific probability vector is very simple. That is, sort the elements of the class-specific probability vector in descending order of probability value and replace the elements by the reciprocal of the rank value.

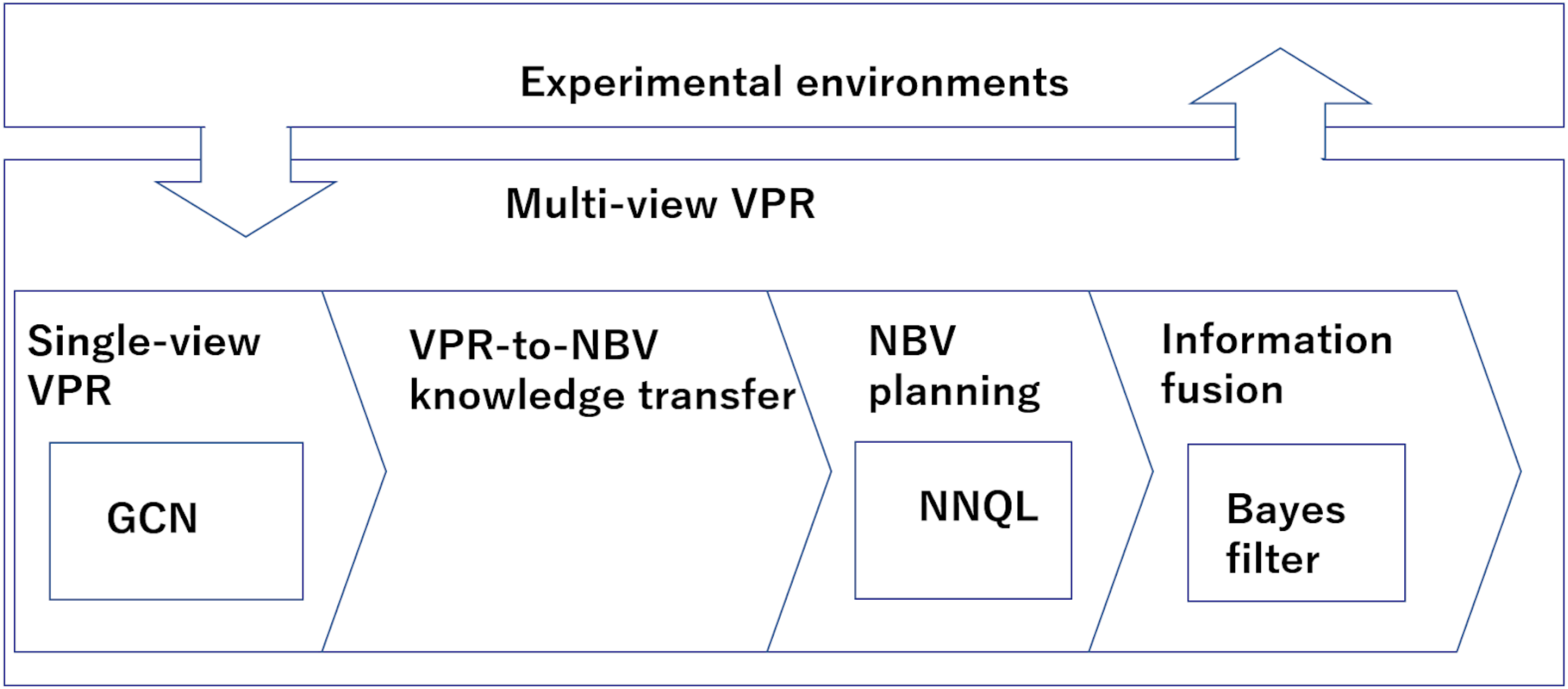


Class-specific probability

Rank

Reciprocal Rank

Next, I will present the experiments.



The active self-localization system employs the passive GCN classifier as the state recognizer.

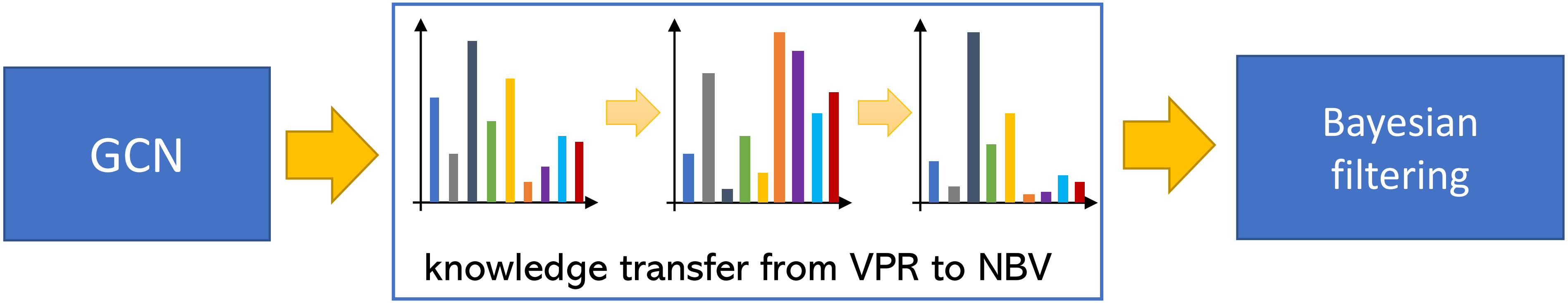
We evaluated both the passive and active self-localization performance.

The proposed system follows multi-view self-localization.

Bayesian filtering is commonly used to validate multi-view information.

In our study, the method was extended to take reciprocal rank features as input.

See the paper for implementation details.



The proposed method was evaluated in a cross-seasonal self-localization scenario. The purpose of the evaluation was to verify whether the GCN-based VPR method could improve performance in both passive and active VPR scenarios.

The publicly available NCLT dataset was used for the experiments. The NCLT dataset is a long-term autonomy dataset acquired using Segway vehicles at the University of Michigan North Campus. While the vehicle moves seamlessly between indoors and outdoors, the vehicle experiences various geometric changes (object placement changes, pedestrians, parked/stopped, etc.) and photometric changes (lighting conditions, shadows, occlusions, etc.).

In particular, we used the four datasets shown in Figure 3. Additionally, the extra season "2012/5/11 (EX)" was used for VPR training. That is, VPR was trained only once in season EX before the self-localization task, after which the learned VPR parameters were commonly used for all training-test season pairs.

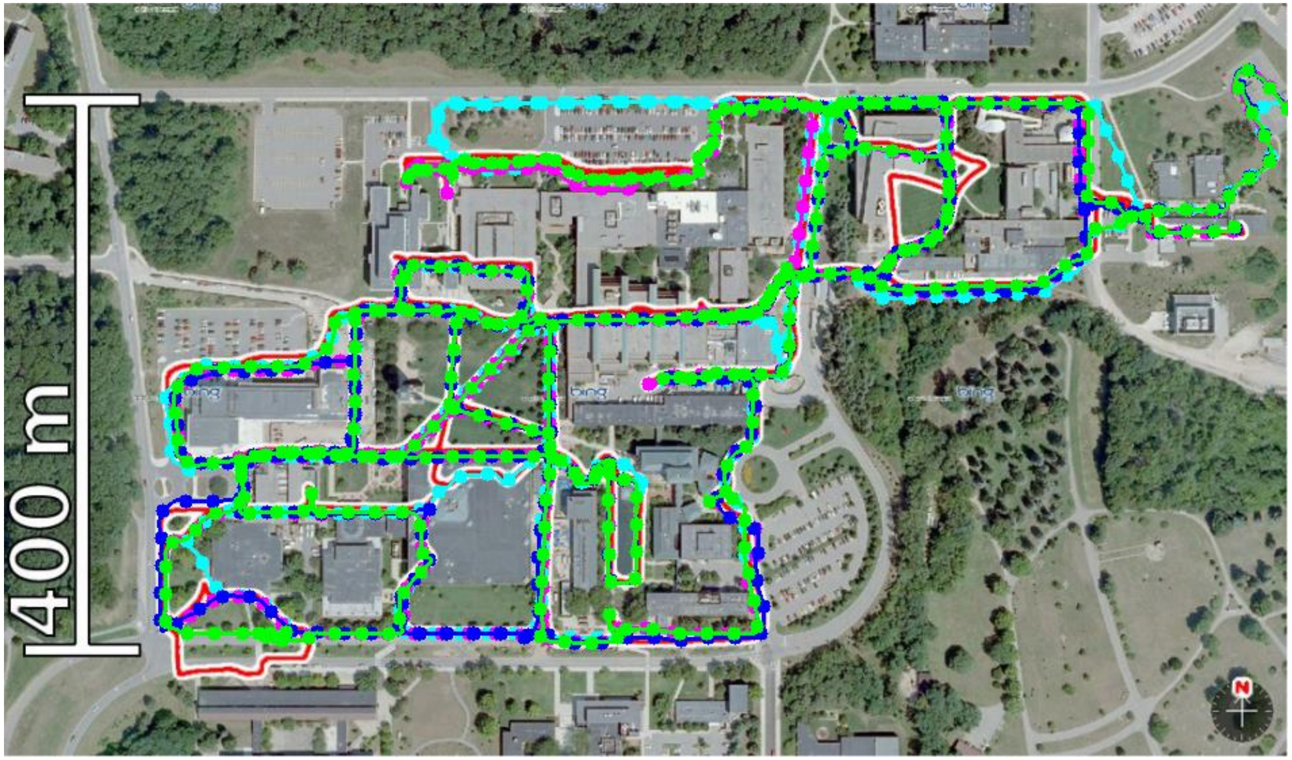


Fig. 3. Experimental environments. The trajectories of the four datasets, “2012/1/22,” “2012/3/31,” “2012/8/4,” and “2012/11/17,” used in our experiments are visualized in green, purple, blue, and light-blue curves, respectively.

Figure 4 shows the scene graph obtained using the proposed scene graph construction procedure. In particular, domain-invariant parts of the input scene (such as buildings and roads) tended to be selected as image domain nodes.

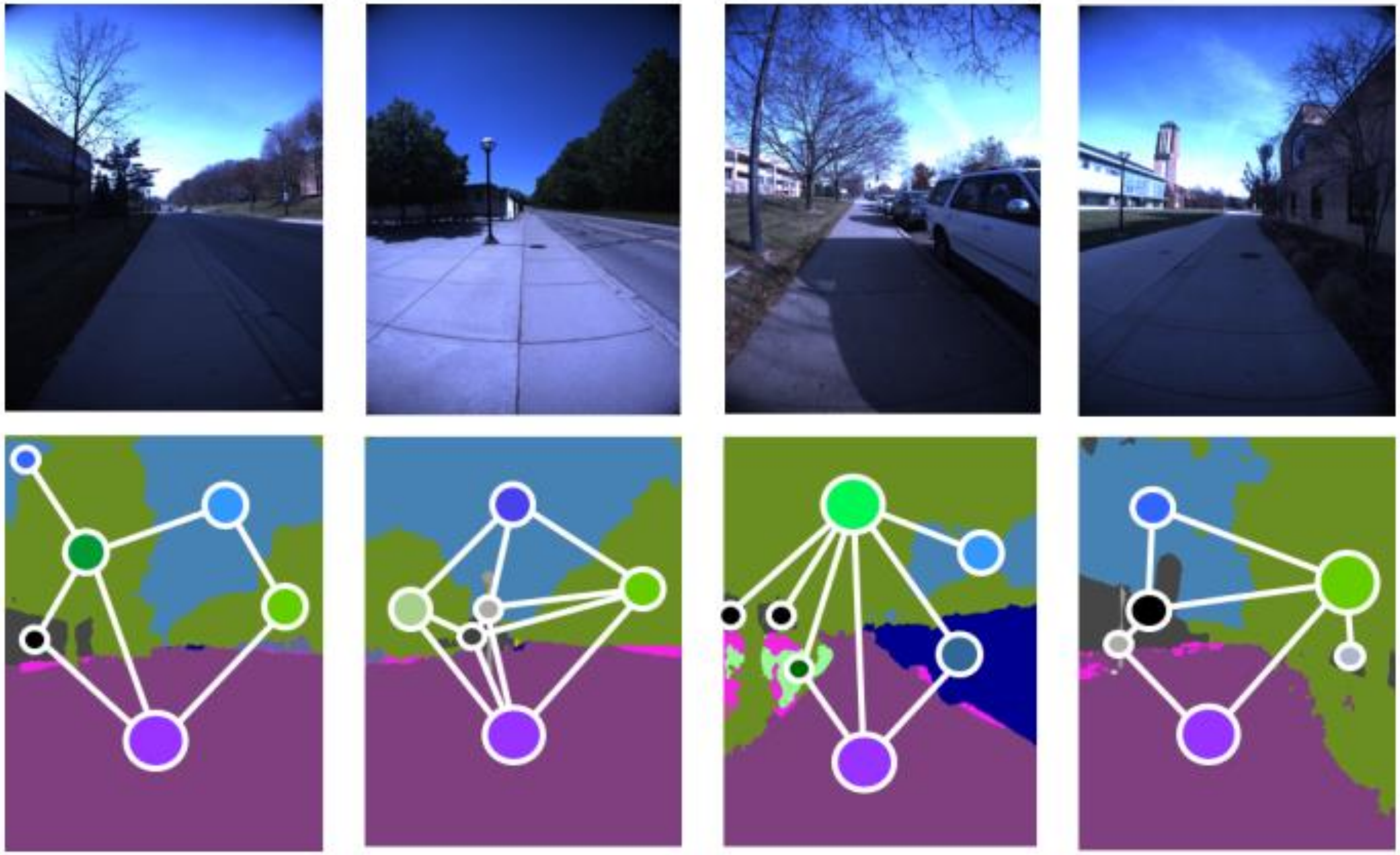


Fig. 4. S2G examples. Top: The input image. Bottom: S2G overlaid on the semantic label image.

Three different VPR methods, GCN, Naive Bayesian Nearest Neighbor (NBNN) and k-Nearest Neighbor (kNN) were evaluated. For details on each method, please refer to the paper. The proposed method is GCN. VPR performance was assessed with top-1 accuracy. The evaluation procedure is as follows.

First, VPR performance was calculated for all views of the query sequence, not just the final view in a multiview VPR task.

The top 1 accuracy at each viewpoint was then calculated from the latest multiview autolocalization (Bayes filter) output based on whether the class with the highest confidence value matched the ground truth.

Ablation studies were performed to observe the effects of individual components, including relative edge features and region-bonding techniques. For details of the ablation studies, please refer to the paper.

Table I lists the result.

In the table, the only difference between "VPR" and "VPR+VP" is whether the view sequence in the multiview VPR was determined randomly or by a proposed view planner.

This result shows that the proposed method has the best performance among the methods considered here.

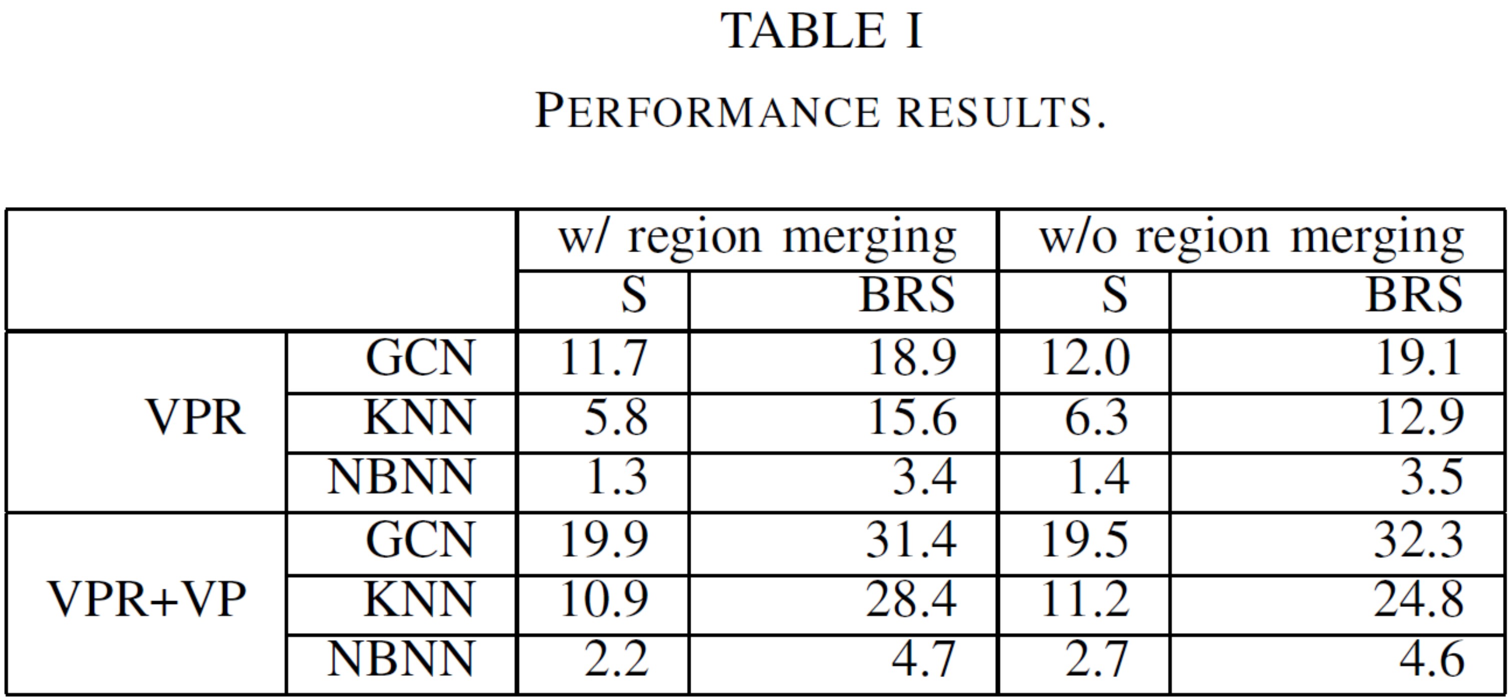


TABLE I

PERFORMANCE RESULTS.

		w/ region merging		w/o region merging	
		S	BRS	S	BRS
VPR	GCN	11.7	18.9	12.0	19.1
	KNN	5.8	15.6	6.3	12.9
	NBNN	1.3	3.4	1.4	3.5
VPR+VP	GCN	19.9	31.4	19.5	32.3
	KNN	10.9	28.4	11.2	24.8
	NBNN	2.2	4.7	2.7	4.6

Figure 5 shows an example of a before and after view of a planned NBV action. Intuitively convincing behavior of the robot was observed.

The pre-movement scene was either an unremarkable scene consisting only of sky, road, and trees (Fig. 1a,b,c) or had a very narrow field of view due to occlusion (Fig. 1d).). After movement, either the landmark object was displayed (Fig. 1a,c) or additional landmarks were displayed (Fig. 1b,d). Such behavior is intuitively appropriate and effective for humans to locate and track landmarks when they get lost (humans look for familiar landmarks). Our approach allows the robot to learn such appropriate step sizes from available visual experience.

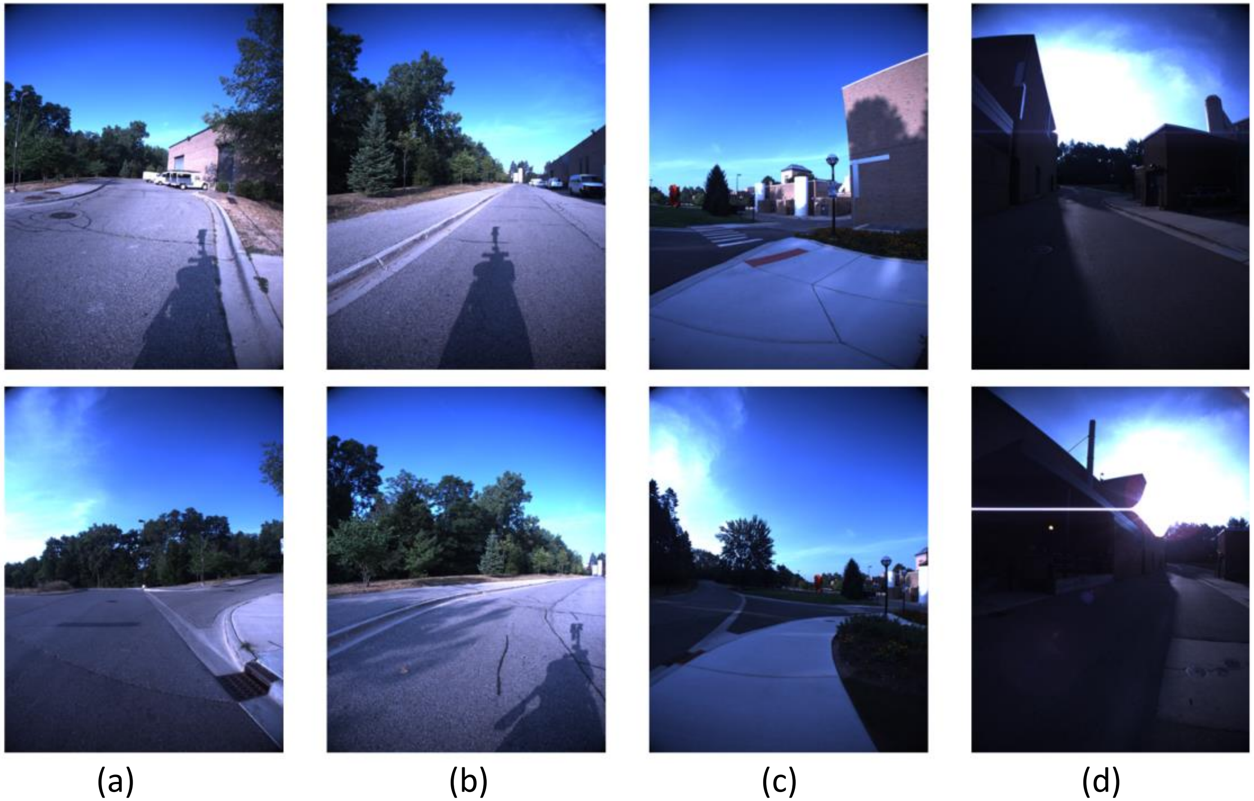


Fig. 5. NBV planning results. In each figure, the bottom and top panels show the view image before and after planned movements, respectively.

Finally, we investigated space costs. The number of nodes per S2G was 7.2 on average. The node descriptor consumed 8-bit per node. The space cost for nodes and edges were 57.8-bit and 12.5bit per S2G, respectively, on average.

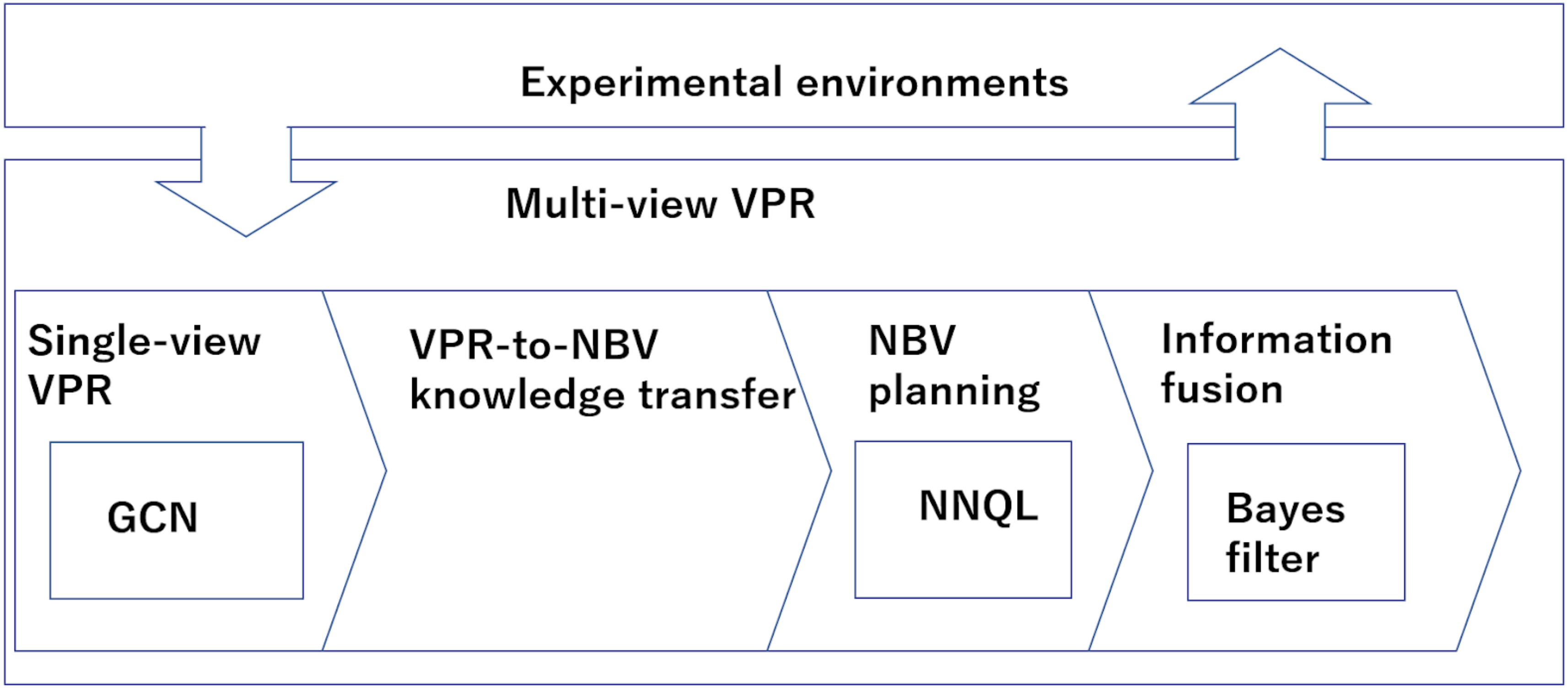
This is significantly lower cost than typical methods based on high-dimensional vectorial features, and than compact variants such as bag-of-words. Notably, the current descriptors were not compressed, that is, they may be further compressed.

number of nodes per S2G	7.2
node descriptor consumed	8-bit
space cost for nodes	57.8-bit
space cost for edges	12.5bit

This is significantly lower cost

A sequential semantic scene graphs (S3G) is promising as an environment model (map) that is robust against changes in viewpoint and appearance. We considered using GCN as a place recognizer for an S3G map. Furthermore, reinforcement learning-based active VPR task was studied by introducing a novel GCN-based graph embedding. The performance was verified by experiments using the NCLT dataset.

In this study, the public NCLT dataset was employed. Such a public dataset is useful for maintaining experimental reproducibility. The NCLT dataset, on the other hand, is passive in nature, with the robot moving along a predefined viewpoint trajectory. Using a simulation environment and a real robot to verify the effectiveness of the proposed method under more conspicuous viewpoint changes are our immediate future tasks.


Experimental environments
Multi-view VPR
Single-view VPR
GCN
VPR-to-NBV knowledge transfer
NBV planning
NNQL
Information fusion
Bayes filter