CityGuessr: City-Level Video Geo-Localization on a Global Scale

Parth Parag Kulkarni[1], Gaurav Kumar Nayak[2], and Mubarak Shah[1]

[1] Center for Research in Computer Vision, University of Central Florida, USA
[2] Mehta Family School of DS & AI, Indian Institute of Technology Roorkee, India
parthparag.kulkarni@ucf.edu; gauravkumar.nayak@mfs.iitr.ac.in; shah@crcv.ucf.edu

Abstract. Video geolocalization is a crucial problem in current times. Given just a video, ascertaining where it was captured from can have a plethora of advantages. The problem of worldwide geolocalization has been tackled before, but only using the image modality. Its video counterpart remains relatively unexplored. Meanwhile, video geolocalization has also garnered some attention in the recent past, but the existing methods are all restricted to specific regions. This motivates us to explore the problem of video geolocalization at a global scale. Hence, we propose a novel problem of worldwide video geolocalization with the objective of hierarchically predicting the correct city, state/province, country, and continent, given a video. However, no large scale video datasets that have extensive worldwide coverage exist, to train models for solving this problem. To this end, we introduce a new dataset, "*CityGuessr68k*" comprising of 68,269 videos from 166 cities all over the world. We also propose a novel baseline approach to this problem, by designing a transformer-based architecture comprising of an elegant "*Self-Cross Attention*" module for incorporating scenes as well as a "*TextLabel Alignment*" strategy for distilling knowledge from textlabels in feature space. To further enhance our location prediction, we also utilize soft-scene labels. Finally we demonstrate the performance of our method on our new dataset as well as Mapillary(MSLS) [38]. Our code and datasets are available here.

Keywords: CityGuessr · geolocalization · Self-Cross Attention · soft-scene labels · TextLabel Alignment

1 Introduction

Geolocalization refers to the process of determining the geographic position of a sample, which can be an image, a video or a text description of a place. If the input sample is strictly visual, i.e., an image or a video, the problem is also termed as Visual Place Recognition (VPR). Geolocalizing images has gained popularity over time, witnessing substantial advancements in the field. In contrast, video geolocalization is currently in its early stages of development. The relevance of video geolocalization today cannot be understated. When the origin of a video is unknown, determining the part of the world the video was recorded in, can assist in a variety of investigative and exploratory applications. The current transformation of social media has resulted in an explosion of video

content, making it a valuable resource. Videos also tend to have more visual information as compared to images, owing to the temporal context that images lack. This makes the problem of video geolocalization even more essential.

There are different levels of granularities in geolocalization problems, right from street identification to worldwide geolocalization, each having its own significance. Image geolocalization has been attempted on both ends of the spectrum [19] [29] [41] [45] [46] [21] [23] [3], with varying levels of success using different approaches catering to the specific problems at hand. Same isn't the case with video geolocalization. There has been some research at the fine-grained level [27] [37] [43], but the same problem at the global level remains largely *unsolved*. Thus, in this work we formulate a unique problem of *worldwide video geolocalization* in an attempt to leverage the information affluence of video domain to address this issue.

Geolocalization in general can be performed in two ways. Retrieval is the more popular approach where a query input is compared with a gallery of known references, which gives the location of the query, provided we are able to find the best match. Although retrieval approaches are more accurate at the fine-grained level, they tend to be computationally expensive and depend heavily on the domains of the queries and references. Any domain shifts tend to have massive repercussions which can snowball quickly into larger issues. Also, constructing a gallery that covers the entire world is not feasible. The second approach, i.e., classification overcomes these limitations. Classification constitutes dividing the region of interest, which in our case is the entire world, into classes, be it in the form of places, or literal partitions of the globe; and identifying the class a sample belongs to. Geolocalization via classification not only decreases compute, but also covers the entire world with ease. As an added advantage, classification can be performed at different hierarchies (city level, state/province level, country level, continent level, etc.), enabling the user to adjust as per their requirements.

Many recent works focusing on the problem of image-geolocalization [39] [28] [21] [23] [3], have proposed classification-based methods for this reason. A general classification pipeline includes an encoder backbone (CNN [17] or transformer [5]-based) to obtain a feature embedding of the input image, and an MLP [13] for class prediction. Previous works also have some additional components or changes in the architecture to aid the classification task, like incorporation of scenes. Keeping this in mind we propose a very unique way of incorporating scenes in our model. Text on the other hand, is relatively unexplored for aiding geolocalization. Intuitively, humans are more likely to identify a location, if they can associate a name to a picture/video of that place which they have previously seen. Consequently, it follows that distilling knowledge from text into a geolocalization model would enhance the model's prediction capability. This motivates us to incorporate text from labels, i.e. city/state/country/continent names during the training procedure by aligning the features of our model to the text embedding of labels without the use of any additional information.

In this paper, we propose a classification-based approach for video geolocalization at global scale. Our objective is to predict the city in which an in-

put video was recorded, and subsequently, the above mentioned hierarchies of state/province, country and continent. Our proposed method comprises of a transformer [5]-based model, with a novel Self-Cross Attention module for scene prediction to assist with the training. We also incorporate soft-labels for computing scene prediction loss during training of our model. A TextLabel Alignment strategy is also implemented for feature enhancement. To the best of our knowledge this is the first attempt to solve this problem, and hence it can serve as a baseline approach for future research.

An obstacle in solving this problem, is the absence of a large scale worldwide dataset for video geolocalization. Existing video geolocalization datasets focus only on specific regions, like BDD [42] in California and New York, USA; KITTI [6] in Karlsruhe, Germany; Brno-Toyota [18] in Brno Czech Republic; and Seqgeo [43] in Vermont, USA. This dense coverage of a limited area works well for retrieval-based geolocalization approaches, but it restricts the data domain to limited parts of the world. To train any model on the global scale, geographic coverage is essential as it exposes the model to a diverse set of locations with variations in environment, infrastructure and salient features instrumental in generalization of the approach. In that context, Mapillary(MSLS) [38], is an image sequence dataset, which covers 30 cities around the world. The geographical coverage, although more than the previously mentioned datasets, is still lacking. The number of sequences in Mapillary is also relatively small, which limits its scope for large scale generalized training. Thus there is a requirement for a large scale global level dataset with a substantial geographical coverage. To this end, we propose CityGuessr68k consisting of $68,269$ videos from 166 cities all over the world. We also provide soft-scene labels for all video samples in the dataset.

Our main contributions can be summarized as follows:

- We formulate a novel problem of worldwide video geolocalization
- To benchmark this new problem, we introduce the first global-scale video dataset named 'CityGuessr68k', containing $68,269$ videos from 166 cities.
- We propose a baseline approach with a transformer-based architecture with two primary components
  - *Self-Cross Attention* module for incorporating scenes (which leverages soft-scene labels for location prediction enhancement)
  - *TextLabel Alignment* strategy for distilling knowledge from textlabels in feature space
- We demonstrate the efficacy of our model with performance results on CityGuessr68k as well as Mapillary(MSLS) datasets.

2 Related Work

Geolocalization can be approached by classification or retrieval. Existing video geolocalization methods are retrieval-based and focus on fine-grained localization. Whereas, all worldwide geolocalization methods are classification-based and cater exclusively to images. We will briefly discuss the relevant works below.

Video Geolocalization It is a young field with very few works, all of which are retrieval-based. Retrieval approaches can either be same-view or cross-view, depending on the domain of the query and reference images. Cross-view has been more popular due to ease of obtaining a reference gallery of satellite images, as compared to ground-view images. Earlier cross-view image models [40] [19] [11] [25] [26] [32] [29] were CNN-based models, while introduction of ViT [5] gave rise to new transformer-based approaches [41] [45]. Although Gonzalo et. al. [34] had proposed a solution for trajectory prediction of a moving camera using Bayesian tracking and Minimum Spanning Trees, until recently, video geolocalization using deep learning was relatively unexplored. GTFL [27] was such a video geolocalization model, which used a hybrid architecture based on VGG [30] along with self-attention, to solve frame-to-frame same-view video geolocalization. Gama-Net [37] extended it to frame-to-frame cross-view video settings by using a hybrid(ResNet [9]/R3D [12]/ViT [5] based) network. Both these methods use BDD100k [42] as a source for query videos, while obtaining reference gallery from other sources. Recently, Seqgeo [43] proposed a clip-to-area model instead of a frame-to-frame approach. However, all these video geolocalization techniques are fine-grained being limited to very few locations. Scaling of these methods to global level is very difficult due to high computation costs and infeasible requirement of a very large reference gallery that covers the whole world.

Worldwide geolocalization It has been image-exclusive since its inception. Weyand et. al [39] first introduced a classification-based approach on the Im2GPS [7] dataset. Vo et. al [36] introduced classification in multiple hierarchies, while on the other hand CPlaNet [28] introduced a combinatorial partitioning technique for combining coarse hierarchies to predict finer ones. Till this point, visual input was the exclusive information available to the model to perform classification. ISNs [21] incorporated scenes, by using three separate encoders for each corresponding scene, depending on whether the image was 'indoor', 'natural' or 'urban'. The concept of hierarchical evaluation, i.e., using coarser hierarchies to refine finer predictions, was also introduced in this paper. Translocator [23] used an additional input of segmentation maps along with images, training twin encoders. Recently, GeoDecoder [3] introduced a completely novel encoder-decoder architecture, which incorporates scenes and hierarchies as queries to the decoder which are attended to by the encoded image features to give seperate embeddings for individual queries, for scene prediction as well as geolocalization.

All recent works attempting to solve this problem train their models on MP-16 [16](except for [28]), while datasets like Im2GPS [7], Im2GPS3k and YFCC4k [36], and YFCC26k [31] are popular for validation. Clark et. al [3] proposed a new validation dataset, GWS15k which is a more balanced dataset with more worldwide coverage. Note that there are no video datasets in this domain. Video datasets like BDD [42], KITTI [6], Brno-Toyota [18] and Seqgeo [43] capture driving and/or walking videos, but all of them are restricted to just one or two geographical regions which make them relevant for training region-scale geolocalization models, but they cannot be used for training geolocalization models at a global scale. While Mapillary(MSLS) [38] dataset is based in multiple

Fig. 1: Sample frames of videos from 22 different countries in the CityGuessr68k dataset. Each quartet represents a continent. The continents in order are, Asia, Africa, Europe, North America, South America and Oceania

cities around the world, it is an image sequence dataset(We do repurpose MSLS to suit our problem and perform experiments on it as described in Section 6). We aim to mitigate this issue, by proposing a global scale dataset which extensively covers different regions of the world, which we call CityGuessr68k. Our dataset can be used as the primary benchmarking dataset for this novel task.

3 CityGuessr68k Dataset

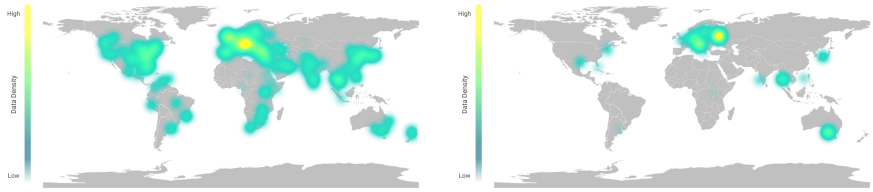
Overview The CityGuessr68k dataset consists of 68,269 first-person driving and walking videos from 166 cities around the world. Fig. 1 shows sample frames from a few videos from varied locations. Each video is annotated with hierarchical location labels, in the form of its continent, country, state/province, city.

Compilation Process The videos in CityGuessr68k, pertaining to different cities, are obtained from the YouTube corpus in the form of long videos of 10-20 minutes. Each video is split into numerous clips with 800-900 frames, out of which 100 frames are sampled. Each sample is then manually annotated with hierarchical labels, based on the city in which the video was recorded. City labels form the most fine-grained hierarchy, from which we go higher to state/province in which the city is located, to the country in which the state/province is located, and finally to the continent in which the country is located. Subsequently, each clip is also further divided into frames, for convenience.

Post-processing As our dataset was collected from the YouTube corpus, and consists of a variety of driving and walking videos from numerous cities, there is a possibility that some videos might contain faces of individuals recorded at the time. To preserve the anonymity of such individuals an extensive post-processing effort was made. All the frames of all videos in the dataset were scanned with RetinaFace [4]. Then, scanned samples were manually inspected, after which, the detected faces were blurred maintaining the privacy of individuals. The entire procedure is described in detail in the supplementary material.

Geographical Distribution Our dataset consists of 68,269 first-person driving/walking videos from 166 cities, 157 states/provinces, 91 countries and 6 continents. Fig. 2 shows the distribution of data samples around the world.

Class Distribution Each city, state/province, country and continent represents a class at their respective hierarchies. Fig. 3 shows the class distribution



(a) Distribution of videos in CityGuessr68k **(b)** Distribution of videos in Mapillary(MSLS)

Fig. 2: Data distribution. A comparison of CityGuessr68k with Mapillary(MSLS) dataset. CityGuessr68k covers more regions of the world, and has a uniform spread around the globe.

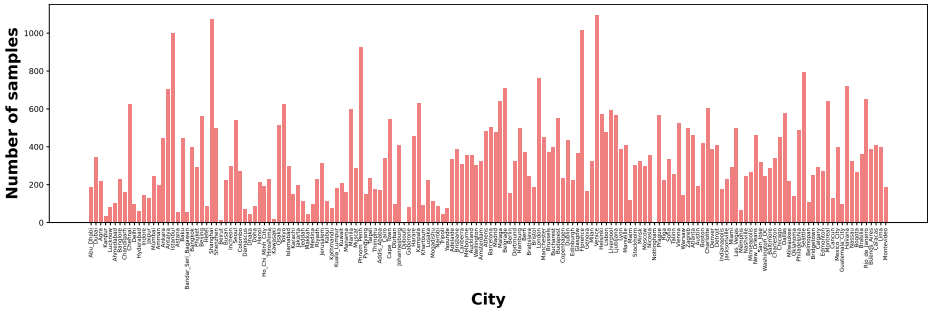
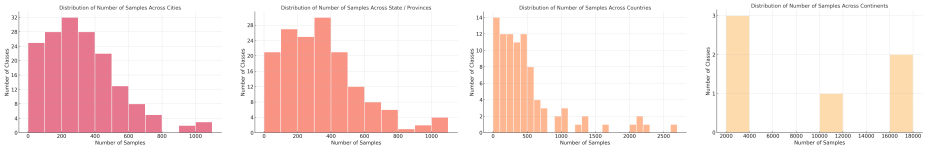


Fig. 3: Class Distribution. Bar chart for number of samples per city class. (please zoom in for clearer class labels)



(a) Histogram of city classes **(b)** Histogram of state classes **(c)** Histogram of country classes **(d)** Histogram of continent classes

Fig. 4: Frequency distribution. Histograms for each hierarchy for a further statistical insight into data distribution of CityGuessr68k.

of our dataset at city level. As discussed above, CityGuessr68k has a good geographical coverage. Along with that, frequency distribution among classes is also relatively even. Fig. 4 shows histograms of classes at all 4 hierarchies. Fig. 4a and 4b show that City and State classes peak in the middle of the graph with mode of both being about 250. This means that the dataset does not have a long tail at these hierarchies. In Fig. 4c, we see that country classes peak very early. It may appear to have a long tail, but the height of subsequent bins is also high. As there are only 6 continents, 3 have less than 10k samples while 1 has around 12k and 2 have around 20k, as observed in Fig. 4d, maintaining a healthy balance.

Video statistics and comparison with Mapillary(MSLS) Each video is divided into frames of resolution 1280x720, which is higher than Mapillary(MSLS)

images. All videos are approximately same in length. The data is organized with every video being contained within an individual folder. Table 1 shows the comparison of our CityGuessr68k dataset with the only other worldwide image sequence dataset, Mapillary(MSLS) [38]. We see that our dataset is $\sim$ 5x larger and spread across more cities around the world.

	Mapillary	CityGuessr68k
Number of samples	14,965*	68,269
Number of cities	30	166
Number of states/provinces	29	157
Number of countries	24	91
Number of continents	6	6
Consistent sequence length	No	Yes
Video sequence	No	Yes
Uniformly Organized	No	Yes
Frame resolution	640x480	1280x720

*Samples with 15 frames or more

Table 1: Comparison with Mapillary(MSLS). This table shows the comparison of our dataset with Mapillary. CityGuessr68k overcomes many shortcomings of the Mapillary dataset.

4 Method

4.1 Problem Statement and Method Overview

Given an input video, our objective is to determine which city in the world, this video was recorded in. This task can also be referred to as Visual Place Recognition. Consequently, we also predict the respective state/province, country, and continent. We approach this problem as a multi-objective classification task. Each video has a corresponding city label, state/province label, country label, and continent label. A model that solves this problem should be able to predict all these labels. Every video also has a unique scene label associated with it. We consider scene recognition and TextLabel Alignment as auxiliary tasks which aid in training our model, details of which is described in Section 4.3 and Section 4.4 respectively.

Method Overview. As shown in Fig. 5, an input video is divided into tubelet tokens, as was first conceptualized in Arnab et. al [1], and passed to a video encoder, which outputs feature embeddings. These embeddings are then input into 4 classifiers, $\mathcal{H}_1$, $\mathcal{H}_2$, $\mathcal{H}_3$ and $\mathcal{H}_4$, representing the city, state/province, country, and continent predictors respectively. The outputs of these classifiers are used to compute the respective losses for each hierarchy. The classifier outputs are also passed on to our 'Self-Cross Attention' module. Its outputs are simultaneously used to compute the scene loss and also passed on to the 'TextLabel Alignment' Module, which aligns these features with textlabel embeddings from a pretrained text encoder via the TLA loss . All the losses (described in Section 4.5) are combined and backpropogated to train the model.

4.2 Encoder Backbone

Encoder backbone is the very first stage of our model. We use VideoMAE [33] to encode features of our input video. VideoMAE is the Masked Auto Encoder(MAE) network, first proposed in He et. al [8], adapted to videos. Originally designed for self-supervised video pretraining for recognition, VideoMAE masks a very high number of tubelets (spatiotemporal tokens), which brings performance improvement while reducing computation significantly. Thus VideoMAE becomes a very good choice for the encoder backbone. Originally VideoMAE authors use a vanilla ViT [5] backbone, and adopt the joint space-time attention [1] [20] to better capture high-level spatiotemporal information in the re-

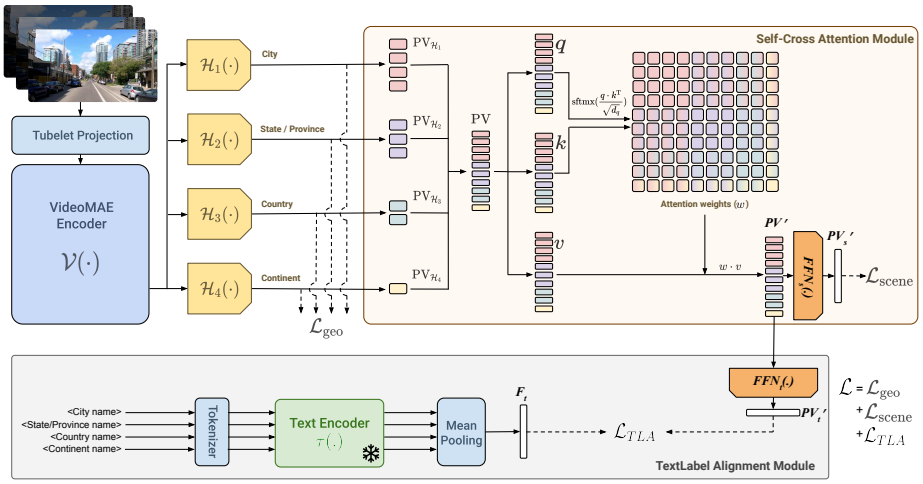


Fig. 5: Schematic Illustration of the proposed Model Architecture. VideoMAE encoder outputs feature embeddings of the input video. The embeddings are then passed into 4 classifiers pertaining to 4 hierarchies. Their predictions are used for computing Geolocalization loss. Simultaneously prediction vectors are input into the Self-Cross Attention module, where vectors of all 4 hierarchies are concatenated and are attended to, by themselves and by each other to generate an intermediate attended vector(PV'). In the attention weights(w), the single colored weights along the diagonal refer to self attention weights, while the gradient double colored weights are the cross attention weights between vectors of those two different hierarchies. PV' is passed simultaneously through FFN_s to generate vector PV_s' for Scene loss computation, and to the TextLabel Alignment module. There, it is passed through FFN_t to generate vector PV_t'. PV_t' is used for TextLabel Alignment with feature embeddings F_t generated by the pretrained text-encoder from the label names of all 4 hierarchies.

maining tokens after masking. The VideoMAE encoder which we have incorporated, is pretrained on Kinetics-400 [14] and is finetuned on our dataset. Encoded features from the backbone are then input to 4 classifiers, one for every hierarchy, which output their respective vectors. These vectors are then passed on to the Self-Cross Attention module, which is described in the following section.

4.3 Scene Recognition

Information pertaining to each hierarchy can influence the scene of a video and vice-versa. Keeping that in mind, we aim to fuse the knowledge from all 4 hierarchies for the identification of the scene pertaining to a video in such a way that the information relevant to each hierarchy enhances the knowledge of other hierarchies as well as its own. This is only possible if there is a means for hierarchies to interact with one another. To this end, we propose Self-Cross Attention module, as described below.

Self-Cross Attention module As mentioned above, scene predictions can be influenced by all hierachies and they need to interact with each other to enhance scene recognition. Our Self-Cross Attention module is designed for that purpose. As shown in Fig. 5, the Self-Cross Attention module takes the output vectors

of all 4 hierarchies ($PV_{\mathcal{H}_1} \in \mathbb{R}^{d_1}$, $PV_{\mathcal{H}_2} \in \mathbb{R}^{d_2}$, $PV_{\mathcal{H}_3} \in \mathbb{R}^{d_3}$, $PV_{\mathcal{H}_4} \in \mathbb{R}^{d_4}$, where d_1, d_2, d_3, d_4 are the number of classes in city, state/province, country and continent hierarchies respectively). These vectors are concatenated to form a vector $PV = concat(PV_{\mathcal{H}_1}, PV_{\mathcal{H}_2}, PV_{\mathcal{H}_3}, PV_{\mathcal{H}_4}) \in \mathbb{R}^d (d = d_1 + d_2 + d_3 + d_4)$. Then the vector is projected into query(q), key(k) and value(v) vectors, which are then used to compute multihead attention [35] (defined as $softmax(\frac{qk^T}{\sqrt{d_q}}).v$, where d_q is the query dimension), $PV' = MHA(PV)$. The attended output is again projected into a vector $\in \mathbb{R}^d$, which is finally passed through a feed-forward network (FFN_s), to gradually reduce the dimension and output the scene vector ($PV'_s = FFN_s(PV') \in \mathbb{R}^{d_s}$, where d_s is the number of scene labels). PV'_s is then used to compute scene loss. Thus, the module essentially performs self attention on output of each hierarchy classifier, as well as cross attention between each pair of hierarchies as depicted in Fig. 5, conceiving the name, *Self-Cross Attention* module. This procedure achieves the intended effect of the outputs from all hierarchies interacting with each other, and with themselves, enhancing location prediction. PV' is also passed on to TextLabel Alignment module.

Scene labels The Self-Cross Attention module, discussed in the previous section, outputs a Scene Vector(PV'_s), which gives a prediction for the scene. To compute scene loss, we require scene labels corresponding to each video sample. For images, assigning scene labels is straightforward, but the same isn't the case with videos. Assigning one scene label to an entire video involves a lot of nuance, as scene might change as we go through all frames. The most trivial way of scene labelling a video can just involve taking the first/middle/random frame and assigning the scene label of that frame to the entire video. That is not a good direction to pursue, as the scene of that frame might not represent the video properly. Majority voting is a more clinical way of approaching this task. This involves assigning a scene label to all frames of a video and assign the scene label which represents the most amount of frames to the video. This is indeed better than the previous approach, but it, too, does not fully capture the complete variation of the video.

Thus, we devise a simple yet effective way of representing the scene label of a video. The concept of soft-labels is interesting, as it captures the detail of a representation. Generally, soft labels are used to train teacher-student distillation models [10]. Instead of assigning a definite class to a sample, soft labels use percentages which capture the probabilities of the sample belonging to each class. Soft labels fit perfectly into our problem setup, as each value of class can be represented as the percentage of frames in the video that belong to a certain class. We use this technique to provide a proper representation of the scene of a video, and assign soft-labels to all videos as the ground truth, which are then used for computing scene loss.

For assigning scene labels to video frames, we use the labels provided by Places2 [44] dataset. We implement a pre-trained scene classification model provided in [44], and get labels for all frames. Then, we convert those into soft labels as mentioned above. We use 16 image scene labels for this work.

4.4 TextLabel Alignment Strategy

Associating a name with a picture/video of a location is helpful for humans to retain and identify the characteristics of that location. Thus location names can be instrumental assets for training a geolocalization model. We pursue this by distilling knowledge from these labels into our model in feature space, through our TextLabel Alignment (TLA) Strategy.

Strategy We compute the textual features of the location labels via a pretrained text-encoder [24]. The objective is to align the features of our model with the generated textlabel features. As is the case with scenes, alignment of textlabel features with combined features of all hierarchies is essential for maximum benefit. Thus we use the features output from the Multihead Attention layer of the Self-Cross Attention module (Section 4.3), before it is passed on to FFN_s. The attended vector (PV' in Section 4.3) is passed through a different feed-forward network (FFN_t) to obtain an output vector($PV'_t = FFN_t(PV') \in \mathbb{R}^{d_t}$, where d_t is the textlabel feature dimension). PV'_t is then used to compute the TLA loss (Eq. 3). Thus, the attended features of all hierarchies are aligned with the textlabel features generated from the text encoder. This helps to distill knowledge from the textlabels to the model features by associating the location's name with the respective video, leading to further enhancement of location prediction.

TextLabel Feature Computation As mentioned above, TextLabel features can be computed in multiple ways. The most trivial way of approaching this is passing only the city names through the text encoder, as association of the city with the input video is the most essential by virtue of it being the finest hierarchy. However, we do lose information from other coarser hierarchies. If all hierarchies are to be incorporated, we can pass each hierarchy label through the text encoder separately and combine the output of the embeddings to obtain the final TextLabel features. The latter option allows us to consider input from all hierarchies without compromising the expression of each hierarchy. A detailed empirical performance analysis with each alternative is shown in Section 5.5.

4.5 Losses

As evident in Fig. 5, our network is trained with three losses, geolocalization loss(L_{geo}), scene loss(L_{scene}) and TLA loss(L_{TLA}). Geolocalization loss is computed as a combination of Cross-Entropy losses(CE) of each hierarchy. Given a video V, L_{geo} can be defined as,

$$L_{geo}(V) = \sum_{i \in \{city, state, country, continent\}} CE(l_i, \hat{l}_i), \tag{1}$$

where, l denotes the ground truth label, while $\hat{l}$ denotes the predicted label. Scene loss is computed as the cross entropy loss between the soft label assigned to a video, and the output vector from the Self-Cross Attention module.

$$L_{scene}(V) = CE(s, PV'_s), \tag{2}$$

where, s denotes the ground truth soft-scene label, and PV_s' denotes the predicted scene vector. TLA loss is computed as the negative cosine similarity between the text features generated from the class labels, and the output vector to be aligned.

$$L_{TLA}(V) = -Cosine\_Similarity(F_t, PV_t'), \tag{3}$$

where, F_t denotes the text features from class labels and PV_t' denotes the output vector. Thus the total loss is

$$L(V) = L_{geo}(V) + L_{scene}(V) + L_{TLA}(V). \tag{4}$$

4.6 Inference

We geolocalize the video V with the outputs of the four classifiers $\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3$ and $\mathcal{H}_4$. $\mathcal{H}_4$ predicts the continent label, $\mathcal{H}_3$ predicts the country label, $\mathcal{H}_2$ predicts the state/province label and $\mathcal{H}_1$ predicts the city label. Now, the predictions for fine-grained hierarchies could be improved with the assistance of the coarser hierarchies. To refine the probabilities of a hierarchy prediction, we can multiply the probabilities of the coarser hierarchies, as they would push the probabilities of the classes in which the coarser hierarchies are most confident. Thus

$$P({C_i}^{\mathcal{H}_1}|V) = P({C_i}^{\mathcal{H}_1}|V) * P({C_j}^{\mathcal{H}_2}|V) * P({C_k}^{\mathcal{H}_3}|V) * P({C_l}^{\mathcal{H}_4}|V) \tag{5}$$

where city ${C_i}^{\mathcal{H}_1}$ is located in state ${C_j}^{\mathcal{H}_2}$, which is located in country ${C_k}^{\mathcal{H}_3}$, which in turn is located in continent ${C_l}^{\mathcal{H}_4}$. Continent probabilities are multiplied into countries and so on. After performing these operations, the predictions of individual probabilities could be independent of each other, or could be codependent on each other. Codependent predictions could be performed by first predicting the most fine-grained hierarchy after multiplying probabilities, and then tracing the hierarchical structure upwards, i.e., once the city is predicted, the state prediction would simply be the state in which the city is located, and so on. Independent predictions are just made by using the individual hierarchy probabilities to determine the most likely class. In our final method, we use codependent predictions, as we found that empirically they are more accurate. More analysis on that is covered in Section 5.6.

5 Experiments, Results and Discussion

This section describes the details of the experiments performed with our model, the data used for those experiments and the training and validation setup in regards to the same.

5.1 Data and metrics

For training and validating our model, we use our newly proposed dataset CityGuessr68k. The dataset is divided into an 80:20 stratified train-test split, all classes being represented in both sets. Thus we train our model on $54,614$ videos and validate on $13,655$ videos. We assess our model's performance using prediction accuracy (top1 accuracy), for each hierarchy. We show additional analysis on top5 accuracy in the supplementary material.

5.2 Training Details

Our model is implemented in pytorch [22]. The input video consists of 15 frames, resized to 224x224. The model is trained on one node of an NVIDIA RTX A6000 GPU. The VideoMAE version used has a ViT-S [5] backbone, and its weights are pretrained on Kinetics-400 [14]. The model is trained for 10 epochs with a batch size of 12. We use the Adam [15] optimizer with a learning rate of 0.001. The Self-Cross Attention module has 2 heads, an embedding dimension of 6 and an FFN that has 6 layers. The TextLabel Alignment strategy utilizes an FFN with 3 layers and the text embedding feature dimension is 512. The video encoding feature dimension is 384.

5.3 Utility of Video data

Video-based geolocalization is more accurate than using single images because videos contain more richer information. We carry out experiments where we replace our video backbone in with its image counterpart (MAE [8]), keeping all other details exactly the same. We test four different settings for the image model. As shown in Table 2 we observe that the "random" setting performs the best. We also observe that the video model outperforms the image models by a large margin. it achieves a 9% improvement over the best image model setting, showing the utility of video data.

Backbone	Setting	City	State	Country	Continent
MAE	First frame	52.1	52.6	55.3	70.4
	Mid frame	48.9	49.3	54.6	69.8
	Last frame	48.1	48.4	53.4	69.3
	Random frame	55.8	56.3	60.8	74.1
VideoMAE video		64.5	64.5	65.9	74.4

Table 2: Comparison of image and video backbones. Comparing performance of MAE and VideoMAE models to demonstrate the necessity of video geolocalization.

Scenes	TLA	City	State	Country	Continent
-	-	64.5	64.5	65.9	74.4
Majority	-	66.9	67.3	72.1	81.1
Soft	-	67.9	68.4	72.4	81.6
Soft	city only	69.1	69.5	73.7	83.1
Soft	all hierarchies	**69.6**	**70.2**	**74.8**	**83.8**

Table 3: Effect of adding Scene recognition and TextLabel Alignment(TLA). Comparing performance of the model with variants of scene labels and TLA

5.4 Scene Recognition

We have introduced a novel technique for incorporating scenes to aid our model training. Self-Cross attention(Section 4.3) between the prediction vectors of hierarchies was conceptualized in an attempt to improve location predictions. We also devised an elegant way of representing scenes with soft labels(Section 4.3) which gives a proper representation of a video scene label making it more suitable for loss computation. Table 3 shows that addition of Self-Cross Attention module certainly helps the model to train better and gives better validation performance. We also showcase our results on two variations of scene labels, one obtained by majority voting and other with soft labels. Comparing their performance, we see that soft labels are more helpful in model training. Note that both models use hierarchical evaluation with codependent predictions.

5.5 Benefit of TextLabel Alignment

We employ our TextLabel Alignment strategy(Section 4.4) to distill knowledge from the names of the locations into our model in the feature space, by aligning textlabel features generated by the pretrained text encoder to the features of our model. We described 2 strategies for computing textlabel features in Section 4.4, from city labels, and from mean of features from all hierarchy labels. Table 3 shows that incorporation of the TextLabel Alignment strategy enhances the features of the model, thus giving a better performance. We showcase our results on both the above described variations. Comparing their performance, we see that using all hierarchies helps the model train better as hypothesized. Note that both models use hierarchical evaluation with codependent predictions.

5.6 Independent v/s Codependent Hierarchical Evaluation

As discussed in Section 4.6, hierarchical evaluation enhances the predictions of the model and as a consequence, improving geolocalization performance. Table 4 shows the same. After multiplying probabilities, coarser hierarchy predictions could be independent, or they could be codependent on finer hierarchies. We evaluate the model using both variations, and Table 4 shows that codependent predictions are better than independent predictions. Note that all the results include Self-Cross Attention module with soft-scene labels and TextLabel Alignment with all hierarchies.

Model	City	State	Country	Continent
w/o hierarchical eval.	69.1	69.6	72.5	79.2
Independent	69.6	69.8	72.5	79.2
Codependent	**69.6**	**70.2**	**74.8**	**83.8**

Table 4: Hierarchical Evaluation. Comparing variants of the model with different types of hierarchical evaluation techniques

Model	City	State	Country	Continent
PlaNet [39]	55.8	56.3	60.8	74.1
ISNs [21]	59.5	59.9	64.1	75.9
GeoDecoder [3]	64.2	64.5	69.5	79.9
Timesformer [2]	60.9	61.4	66.1	78.4
VideoMAE [33]	64.5	64.5	65.9	74.4
Ours	**69.6**	**70.2**	**74.8**	**83.8**

Table 5: Comparison of our method with baselines and state-of-the-art methods

5.7 Comparison with State-of-the-art

As stated in Section 5.1, we show the validation performance of our model on 13,655 videos from CityGuessr68k. As no worldwide video geolocalization methods exist, we compare our model to the baselines with TimesFormer [2] and VideoMAE [33] encoders, along with the relevant state-of-the-art image geolocalization methods. For image models, we use the random frame setting, as it performed the best for the image MAE [8] baseline(Section 5.3). Hierarchy classifiers are included for all models, and everything else is kept the same as per specifications mentioned in Section 5.2. Table 5 shows the results of our model on our dataset. Our model is able to achieve a 69.6% top1 accuracy on City prediction, i.e., the most fine-grained hierarchy. Our model showcases an improvement of $\sim$ 6% highlighting the significance of our modules. Our model also shows an improvement in the coarser hierarchies with an $\sim$ 6% jump in state/province prediction, an $\sim$ 5% improvement in country and a $\sim$ 4% in continent prediction.

6 Performance on Mapillary(MSLS)

Mapillary(MSLS) [38] is an image sequence dataset, with sequences of varying length from 30 cities around the world. As discussed in Section 3, Mapillary does have some shortcomings. Also, from Fig. 2b, we observe that Mapillary does not cover a lot of locations around the world. We propose CityGuessr68k to address all these concerns. However, we validate the effectiveness of our model by also showing its performance on Mapillary.

Data preparation Due to its design, there are a lot of steps involved in making the Mapillary dataset compatible with our model. Mapillary is an image sequence dataset spread across multiple folders and subfolders. To its credit, it was originally collected for training image sequence retrieval models and thus every city has some query sequences and some database sequences. As we are performing a classification task, we do not require seperate query and database sequences. Thus we decided to combine sequences from both for our purposes. We also further reformat, filter and split the dataset such that it is compatible with our problem configuration. The procedure of data preparation is further detailed in the Supplementary material.

Experiments and Results After the filtering and train-test split, we had 9049 train sequences and 2271 validation sequences. We assess the performance again using prediction accuracy (top1 accuracy). Training parameters were kept exactly the same as described in Section 5.2. Table 6 shows the validation results of our model with Self-Cross Attention module trained with soft-scene labels, TextLabel Alignment strategy with all hierarchies and codependent hierarchical evaluation. We compare the performance of our model against the VideoMAE baseline. We see that there's a 5% jump in top1 accuracy on city prediction, as well as significant improvements in coarser hierarchies as well. This shows that our model trains and performs well on other datasets and thus can be generalized for this task across different data distributions.

Model	City	State	Country	Continent
VideoMAE	67.6	67.6	68.2	81.9
Ours	**72.8**	**72.8**	**73.2**	**88.1**

Table 6: Performance comparison on Mapillary(MSLS) dataset. Our method compared with the VideoMAE baseline

7 Conclusion

In this paper, we formulated a novel problem of worldwide video geolocalization. As there is no large scale dataset to tackle this challenging problem, we introduced a new global level video dataset, CityGuessr68k, containing 68,269 videos from 166 cities. We also proposed a baseline approach which consists of a transformer-based architecture with a Self-Cross Attention module for incorporating an auxiliary task of scene recognition with soft-scene labels as well as a TextLabel Alignment strategy to distill knowledge from location labels in feature space. We demonstrated the efficacy of our method on our dataset as well as on Mapillary(MSLS) dataset. As a future direction, we plan to explore the generalizability of the combination of Self-Cross Attention module and TextLabel Alignment to other hierarchical video classification tasks.

Acknowledgements

This work was supported in parts by the US Army contract W911NF-2120192 and National Geospatial-Intelligence Agency(NGA) Award # HM0476-20-1-0001. We would like to extend our gratitude to all the reviewers for their valuable suggestions. We like to thank high school students Emily Park, Megan Shah and Vikram Kumar for their contributions towards collecting data and to Robert Browning and Dr. Krishna Regmi for mentoring them. We would also like to thank Vicente Vivanco Cepeda, Akashdeep Chakraborty, Prakash Chandra Chhipa, Manu S Pillai and Brian Dina for their contributions towards the dataset and insightful discussions.

References

1. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C.: Vivit: A video vision transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6836–6846 (2021) 7
2. Bertasius, G., Wang, H., Torresani, L.: Is space-time attention all you need for video understanding? In: ICML. vol. 2, p. 4 (2021) 13
3. Clark, B., Kerrigan, A., Kulkarni, P.P., Cepeda, V.V., Shah, M.: Where we are and what we're looking at: Query based worldwide image geo-localization using hierarchies and scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23182–23190 (2023) 2, 4, 13
4. Deng, J., Guo, J., Ververas, E., Kotsia, I., Zafeiriou, S.: Retinaface: Single-shot multi-level face localisation in the wild. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5203–5212 (2020) 5
5. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020) 2, 3, 4, 7, 12
6. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE conference on computer vision and pattern recognition. pp. 3354–3361. IEEE (2012) 3, 4
7. Hays, J., Efros, A.A.: Im2gps: estimating geographic information from a single image. In: 2008 ieee conference on computer vision and pattern recognition. pp. 1–8. IEEE (2008) 4
8. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16000–16009 (2022) 7, 12, 13
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016) 4
10. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015) 9
11. Hu, S., Feng, M., Nguyen, R.M., Lee, G.H.: Cvm-net: Cross-view matching network for image-based ground-to-aerial geo-localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7258–7267 (2018) 4

12. Ji, S., Xu, W., Yang, M., Yu, K.: 3d convolutional neural networks for human action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(1), 221–231 (2013). https://doi.org/10.1109/TPAMI.2012.59 4
13. Kawaguchi, K.: A multithreaded software model for backpropagation neural network applications. The University of Texas at El Paso (2000) 2
14. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017) 8, 12
15. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 12
16. Larson, M., Soleymani, M., Gravier, G., Ionescu, B., Jones, G.J.: The benchmarking initiative for multimedia evaluation: Mediaeval 2016. IEEE MultiMedia **24**(1), 93–96 (2017) 4
17. LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., Jackel, L.: Handwritten digit recognition with a back-propagation network. In: Touretzky, D. (ed.) Advances in Neural Information Processing Systems. vol. 2. Morgan-Kaufmann (1989), https://proceedings.neurips.cc/paper_files/paper/1989/file/53c3bce66e43be4f209556518c2fcb54-Paper.pdf 2
18. Ligocki, A., Jelinek, A., Zalud, L.: Brno urban dataset-the new data for self-driving agents and mapping tasks. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 3284–3290. IEEE (2020) 3, 4
19. Liu, L., Li, H.: Lending orientation to neural networks for cross-view geo-localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5624–5633 (2019) 2, 4
20. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10012–10022 (2021) 7
21. Muller-Budack, E., Pustu-Iren, K., Ewerth, R.: Geolocation estimation of photos using a hierarchical model and scene classification. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 563–579 (2018) 2, 4, 13
22. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems **32** (2019) 12
23. Pramanick, S., Nowara, E.M., Gleason, J., Castillo, C.D., Chellappa, R.: Where in the world is this image? transformer-based geo-localization in the wild. In: European Conference on Computer Vision. pp. 196–215. Springer (2022) 2, 4
24. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021) 10
25. Regmi, K., Borji, A.: Cross-view image synthesis using geometry-guided conditional gans. Computer Vision and Image Understanding **187**, 102788 (2019) 4
26. Regmi, K., Shah, M.: Bridging the domain gap for ground-to-aerial image matching. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 470–479 (2019) 4
27. Regmi, K., Shah, M.: Video geo-localization employing geo-temporal feature learning and gps trajectory smoothing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12126–12135 (2021) 2, 4

28. Seo, P.H., Weyand, T., Sim, J., Han, B.: Cplanet: Enhancing image geolocalization by combinatorial partitioning of maps. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 536–551 (2018) 2, 4
29. Shi, Y., Yu, X., Campbell, D., Li, H.: Where am i looking at? joint location and orientation estimation by cross-view matching. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4064–4072 (2020) 2, 4
30. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014) 4
31. Theiner, J., Müller-Budack, E., Ewerth, R.: Interpretable semantic photo geolocation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 750–760 (2022) 4
32. Toker, A., Zhou, Q., Maximov, M., Leal-Taixé, L.: Coming down to earth: Satellite-to-street view synthesis for geo-localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6488–6497 (2021) 4
33. Tong, Z., Song, Y., Wang, J., Wang, L.: Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. Advances in neural information processing systems **35**, 10078–10093 (2022) 7, 13
34. Vaca-Castano, G., Zamir, A.R., Shah, M.: City scale geo-spatial trajectory estimation of a moving camera. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1186–1193. IEEE (2012) 4
35. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017) 9
36. Vo, N., Jacobs, N., Hays, J.: Revisiting im2gps in the deep learning era. In: Proceedings of the IEEE international conference on computer vision. pp. 2621–2630 (2017) 4
37. Vyas, S., Chen, C., Shah, M.: Gama: Cross-view video geo-localization. In: European Conference on Computer Vision. pp. 440–456. Springer (2022) 2, 4
38. Warburg, F., Hauberg, S., Lopez-Antequera, M., Gargallo, P., Kuang, Y., Civera, J.: Mapillary street-level sequences: A dataset for lifelong place recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2626–2635 (2020) 1, 3, 4, 7, 14
39. Weyand, T., Kostrikov, I., Philbin, J.: Planet-photo geolocation with convolutional neural networks. In: European Conference on Computer Vision. pp. 37–55. Springer (2016) 2, 4, 13
40. Workman, S., Souvenir, R., Jacobs, N.: Wide-area image geolocalization with aerial reference imagery. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3961–3969 (2015) 4
41. Yang, H., Lu, X., Zhu, Y.: Cross-view geo-localization with layer-to-layer transformer. Advances in Neural Information Processing Systems **34**, 29009–29020 (2021) 2, 4
42. Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., Darrell, T.: Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2636–2645 (2020) 3, 4
43. Zhang, X., Sultani, W., Wshah, S.: Cross-view image sequence geo-localization. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 2914–2923 (2023) 2, 3, 4

44. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. IEEE transactions on pattern analysis and machine intelligence **40**(6), 1452–1464 (2017) 9
45. Zhu, S., Shah, M., Chen, C.: Transgeo: Transformer is all you need for cross-view image geo-localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1162–1171 (2022) 2, 4
46. Zhu, S., Yang, L., Chen, C., Shah, M., Shen, X., Wang, H.: R2former: Unified retrieval and reranking transformer for place recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19370–19380 (2023) 2