Multi-scale Attention-Based Inclination Angles Estimation for Panoramic Camera

Yuhao Shan[1], Heyu Chen[1], Jiaying Zhang[1], Shigang Li[2], Jianfeng Li[1*]
[1]Southwest University, Chongqing, China
[2]Hiroshima City University, Hiroshima, Japan
shanyuhao@swu.edu.cn, popqlee@swu.edu.cn
*corresponding author

Abstract

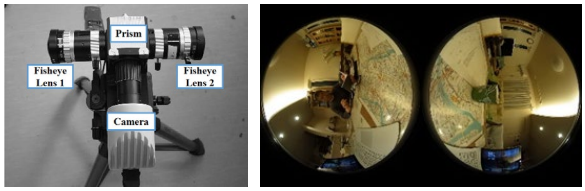
Images taken by panoramic cameras in the upright posture can give viewers a better sense and make the downstream panoramic image-based computer vision tasks easier. To estimate the inclination angles of panoramic camera, we proposed a simple but elegant panoramic image-based network, which combines the advantages of geometry-based and deep-learning-based methods. First, a backbone network with five down-sampling layers is designed to focus on the local distortion features. Then, since non-upright panoramic images have highly uniform geometric distortion for the same camera inclination angles, a multi-scale attention module is proposed for the first time, which can weigh each pixel on the feature maps of the backbone network and allows the network to focus on the global and shallow geometric features. Moreover, apart from angle loss, pixel-level image loss is introduced in our network for the inclination angles estimation task to allow the network to compensate for pixel deviations during training. The experiments show that our method overcomes other leading state-of-the-art methods in this field.

1. Introduction

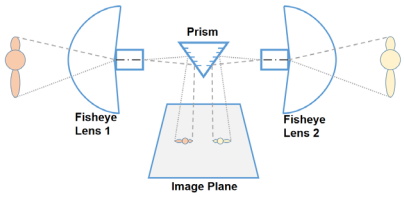
Panoramic images and videos are new forms of multimedia generated using panoramic cameras that can provide a more immersive experience due to their capability of providing a 360° view. Upright panoramic images can give users a better sense of space and make the downstream panoramic image-based computer vision tasks easier, such as image segmentation, object detection, depth estimation, scene understanding, robot navigation and three-dimensional (3D) reconstruction [1-6, 17-23], etc.

Fig. 1 shows the description of non-upright imaging problem based on panoramic camera structure and panoramic imaging principle. In essence, a panoramic camera consists of a pair of back-to-back fisheye lenses with a hemispherical field-of-view, and a central prism. As shown in Fig. 1 (a & c): When incoming light enters the fisheye lenses, it will be refracted to smaller exit angles than its incident angles, and then reflected by the prism and imaged on a plane detector with limited size. Fig.1 (b) shows two hemispherical views captured by these two fisheye lenses, which are called the raw panoramic image. Since constant upright imaging using the camera cannot be guaranteed, the attitude of the camera will affect the angle of the incident light and consequently the imaging results.

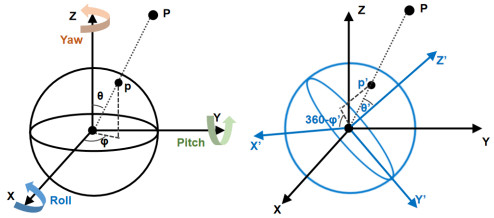
As shown in Fig. 1 (d & e): When the panoramic camera



(a) Panoramic camera (b) Raw image



(c) Internal structure of a panoramic camera



(d) Spherical projection models: (left) projection model of an upright camera; (right) projection model of a tilted camera.

(e) Equirectangular images: (left) image obtained with the camera at an upright posture; (right) image obtained at the same position with an inclined camera.

Figure 1: Description of non-upright imaging problem based on panoramic camera structure and panoramic imaging principle.

is tilted, the camera's coordinate system rotates from XYZ to X'Y'Z', and the imaging position in the 2D projection image of any 3D space point P will change depending on the pitch angle and yaw angle of the camera, that is to say, the geometric deformation of the scene in the panoramic image is closely related to the inclination angles of the camera; Since equirectangular image is the most popular 2D projection image of panoramic imaging and can be generated using the spherical projection model, the inputs of the proposed panoramic camera inclination angles estimation network is this kind of images. The outputs of the proposed network are angles needed to rotate the non-upright panoramic image to the upright image, that is, the two inclination angles of the camera during imaging.

To estimate the inclination angles of the panoramic camera: ***Conventional geometry-based methods*** usually detect vanishing points using orthogonal lines in images based on the assumption of a Manhattan world or an Atlanta world [7-10, 26-28]. Under ideal conditions, these methods are intuitive and very effective, but the assumptions are not always appropriate; In some scenes, there are not so many orthogonal lines, such as dense forests, empty deserts and grasslands, and these types of methods are easily disturbed by noise in the process of detecting geometric information. ***Deep learning-based methods*** [11-16] have shown strong robustness to noise due to the powerful learning ability of neural networks. Most of these methods feed the entire panoramic image into a network with many layers, and only use the information from the small-sized feature maps of the last layer to estimate the rotation angles of the camera. However, the features from the deeper layers usually contain more abstract coarse-grained information about the specific contents of a scene, while the features from shallow layers contain more globally geometrically fine-grained information. ***Humans*** typically use geometrically information, such as lines and the outline of the objects in the image, to correct non-upright images. Fig. 2 shows two panoramic images taken by panoramic camera at the same inclination angles, we can see that although the scenes are different, the horizontal and vertical lines at the same position in different images have the same distortion pattern, such as the distortion pattern of the horizon marked with red lines. Therefore, the use of only deep features is insufficient, because they ignore an important characteristic of non-upright imaging, namely that images contain geometric information that is essential for the estimation of camera's inclination angles. Additionally, most existing methods rely only on inclination angle-related loss functions to constrain the panoramic camera's inclination angles estimation, whereas the non-upright camera's attitude is closely related to the position of each pixel in the image.

Figure 2: Upright equirectangular images and rotate them by the same pitch and roll angles (-40°, 30°).

To overcome these shortcomings, we proposed a simple but elegant network to achieve the panoramic camera inclination angle estimation in the range of ±90 degrees: A concise backbone network with only five down-sampling layers is designed to focus on the local distortion features, and the following novel characteristics are introduced.

- A multi-scale shallow geometric feature attention mechanism is proposed to strengthen the network's attention to the geometric deformation characteristics caused by the camera inclination.
- Apart from angle loss, pixel-level image loss is introduced into the camera inclination estimation task for the first time to improve convergence accuracy of the network.
- The comparison experiment with several state-of-the-art methods shows that the proposed method achieves very competitive results in this field.

2. Related Works

Camera inclination estimation is a basic problem in computer vision. We divide previous panoramic camera inclination estimation methods into two categories: conventional geometry-based methods and deep-learning-based methods.

2.1. Conventional Geometry-based Methods

Tsubasa Goto et al. [8] proposed a line-based global location method for panoramic camera based on the Manhattan world assumption, and used the spherical Hough transform to detect lines in the image and obtain the three principal directions of the panoramic camera. Compared with the Manhattan world assumption, which defines three orthogonal directions, the Atlanta world assumption is more suitable for general scenarios: it assumes that the scene consists of a vertical direction and a set of horizontal directions orthogonal to that vertical direction. Jinwoong Jung et al. [7, 10] proposed an upright posture adjustment method of panoramic camera based on the Atlanta world assumption, and used horizontal and vertical lines in the scene to formulate a cost function for camera inclination estimation. However, as mentioned before, these assumptions are not always valid. In some

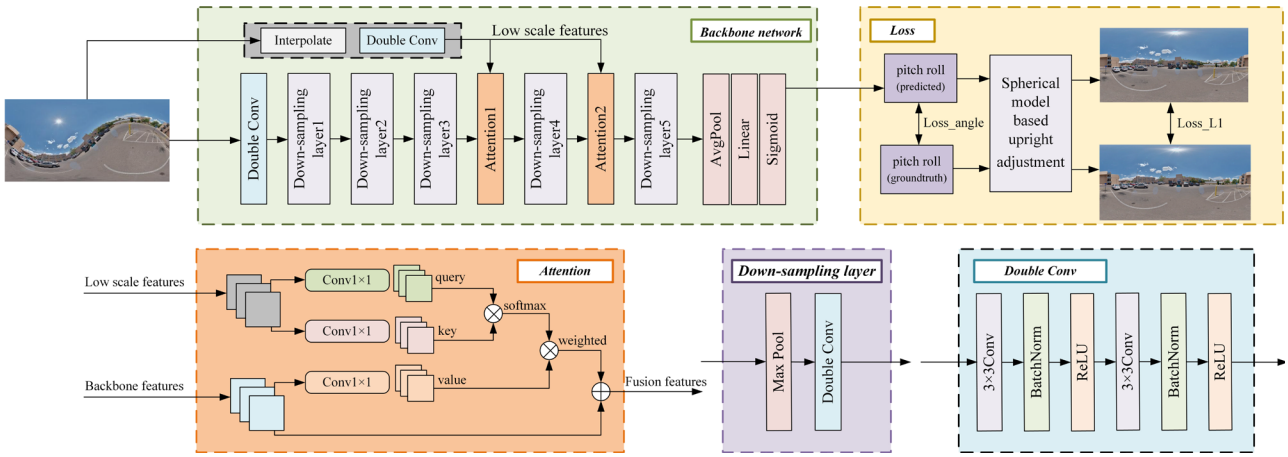


Figure 3: Overview of the proposed multi-scale attention-based inclination angles estimation method for panoramic camera.

scenes, there are not so many orthogonal lines. In addition, this type of method calculates the vanishing points based on the detected lines, and then uses the vanishing points to correct the non-upright camera, the accuracy of the correction is easily affected by the noise present during the geometric information extraction process.

2.2. Deep-learning-based Methods

Raehyuk Jung et al. [11, 12] proposed two panoramic camera inclination angle estimation methods based on convolutional neural network (CNN) and graph convolutional neural network. Shan et al. [14] discussed three 2D representations of panoramic images, and proposed a CNN-based coarse-to-fine inclination angle estimation method for panoramic camera based on a discrete spherical image (DSI): This method first estimates and adjusts the inclination angle of the camera to the range of ±10°, and then corrects it to the range of ±5°, and finally to the range of ±1°. These training-based methods do not rely on any assumptions, so they are suitable for many common scenarios and show strong robustness to noise interference. Although these methods demonstrate the powerful learning ability of neural networks, they only use the features of the last layer to achieve camera inclination angle estimation, without taking into account the shallow geometric information in the image to guide the network's training. In addition, the method proposed by Shan et al. [14] involves training the CNN using DSI with smaller deformations, but this representation has to be obtained through complex preprocessing, and it is not as intuitive as the most common equirectangular image. Benjamin Davidson et al. [15] proposed a modified GSCNN [29] for the camera inclination estimation, where geometric clues (e.g., vanishing points) and semantic clues (e.g., doors) were used for network training. Chen et al. [16] proposed an end-to-end generative adversarial network to generate upright panoramic images.

3. Proposed Methods

We present the detail of the proposed multi-scale attention-based panoramic camera inclination angle estimation method in this section. Firstly, the inclination angle estimation task for panoramic camera will be introduced in Section 3.1; Then, as the overview shown in Fig. 3, our network architecture can be split into three main aspects: the backbone network for extracting local distortion features will be described in Section 3.2, the multi-scale attention module for focusing on global and shallow geometric features will be described in Section 3.3, and the joint loss function of fused image loss and angle loss will be introduced in Section 3.4.

3.1. Camera Inclination Estimation Task

The proposed method focuses on the estimation of inclination angles in the range of 180° (±90°) as in practice cameras are rarely tilted beyond this range. The inputs of the network are non-upright panoramic images, and the outputs of the network are the estimation of the corresponding pitch and roll angles of the camera during imaging. The equirectangular images with a size of 256×512 were utilized in our method for two reasons: firstly, considering computational efficiency, high resolution is not necessary for estimating the inclination; secondly, this resolution or even lower has been commonly employed in previous studies. By utilizing the estimated angles, the camera's attitude or the image of any resolution can be promptly corrected to an upright attitude.

3.2. Backbone Network

Recall that humans typically use geometrically information to correct non-upright images, and the features from shallow layers of the neural network contain more globally geometrically fine-grained information. Inspired

by this, the backbone network should not be too deep, and the input images should not be too down-sampled.

As shown in the upper left part of Fig. 3, our backbone network consists of five cascading down-sampling layers, each containing a 2×2 maximum pooling layer and a double convolution layer. The double convolution layer does not change the size of the input feature maps, but only refines the features further. Specifically, the image is first input to a double convolution layer for preprocessing, and 32 feature maps are obtained. Then, these feature maps are input into down-sampling layers, where the number of feature maps from the first to the fifth down-sampling layer is 64, 128, 256, 512 and 1024, respectively. Next, feature maps from the fifth down-sampling layer are input to an average pooling layer, and a vector of length 1024 is obtained. Finally, the linear layer projects this vector as a vector of length 2, and the ensuing sigmoid nonlinear activation function converts this vector's two values between 0 and 1. These two values are the estimated normalized rotation angles of the camera, i.e., the pitch and roll angles.

The backbone network can be used to extract the local distortion features of a specific scene in the panoramic image. As explained previously, the accuracy of this approach will be limited, as it takes into account only local distortion features and ignores geometric and pixel-level information.

3.3. The Multi-scale Attention Module

Although the proposed backbone network is not deep, some loss of shallow global geometric features will still occur. It is necessary to introduce a multi-scale attention module for shallow geometric features to guide network to pay more attention to the important areas during training, so that the network can learn faster and more efficiently.

As shown in the upper left part of Fig. 3, attention modules of different scales are inserted behind the third and fourth down-sampling layers of the backbone network. The attention modules perform global geometric distortion attention weighing on feature maps output from the two down-sampling layers. The motivation for adopting a multi-scale attention mechanism is that multi-scale feature maps allow the network to simultaneously extract the geometric information of small and large objects during training, such as the edges of ceiling and walls, or the edges of doors and tables. This information is exactly what human beings pay attention to when adjusting a non-upright image. As shown in the lower left part of Fig. 3, the attention module has two inputs with the same dimension: one is feature maps from the backbone network; the other is the low-scale shallow feature maps, which are extracted from the resized original image using only one double convolution layer. The query matrix ($\boldsymbol{Q}$) and key matrix ($\boldsymbol{K}$) are extracted from the low-scale feature maps using two different 1×1 convolutions, respectively. The value matrix ($\boldsymbol{V}$) is extracted from the feature maps of the backbone network through 1×1 convolution. By calculating the similarity between $\boldsymbol{Q}$ and $\boldsymbol{K}$, an attention map is obtained which indicates which geometric information is important in the panoramic image. Then, the attention map is normalized through the SoftMax activation function so that the sum of its values is equal to 1, and the spatial attention weight map is obtained. Finally, the latter is used to weigh each pixel in $\boldsymbol{V}$ by matrix multiplication, and the attention feature maps are obtained. The fusion of attention feature maps and backbone feature maps can be expressed as follows:

$$\boldsymbol{Fusion\ features} = \boldsymbol{ReLU}(\boldsymbol{backbone}_{\boldsymbol{fms}} + \boldsymbol{Atten}_{\boldsymbol{fms}}) \quad (1)$$

where $Atten_{fms}$ represents the attention feature maps, $backbone_{fms}$ represents the backbone feature maps. The nonlinear activation function ReLU increases the sparsity of the feature maps and the nonlinearity of the network. The fused feature maps are then input into subsequent down-sampling layers. It is worth mentioning that in order to avoid the branches of the network interfering too much with the learning of the backbone network, only the branches are used as the attention mechanism and the calculated attention weight is introduced without changing the dimensions of backbone network features.

3.4. Loss Functions

As shown in the upper right part of Fig. 3, different from the previous works which only used the angle-dependent loss function [11, 14, 15], in this work the fusion of pixel-wise image loss with angle loss is proposed to constrain the convergence of the network.

The angle loss is defined as the smooth L1 loss between ground truth angles and the estimated angles, that is:

$$\boldsymbol{L_{ang}} = \begin{cases} \boldsymbol{0.5 \times (L1_{angle})^2}, & \boldsymbol{\left|L1_{ang}\right| < 1} \\ \boldsymbol{\left|L1_{angle}\right| - 0.5}, & \boldsymbol{otherwise} \end{cases} \quad (2)$$

where $L1_{ang}$ is the L1 loss between ground truth angles and the estimated angles.

The image loss is the L1 loss between the ground truth upright image and the image after upright adjustment with the angles estimated by the network, that is

$$\boldsymbol{L_{img}} = \boldsymbol{\sum_{i=1}^{256}\sum_{j=1}^{512}\left|Img_{gt}(i,j) - Img_{adj}(i,j)\right|} \quad (3)$$

where Img_{gt} is the ground truth upright image, Img_{adj} is the image adjusted according to the inclination angles estimated by the network, and $\boldsymbol{i}$ and $\boldsymbol{j}$ are the row index and column index of pixels in the image.

Although the principle seems very simple, obtaining Img_{adj} quickly is a difficult problem in network training. Fig. 4 shows how to obtain Img_{adj} in our method, it's a unit-spherical-model-based method: We first project the

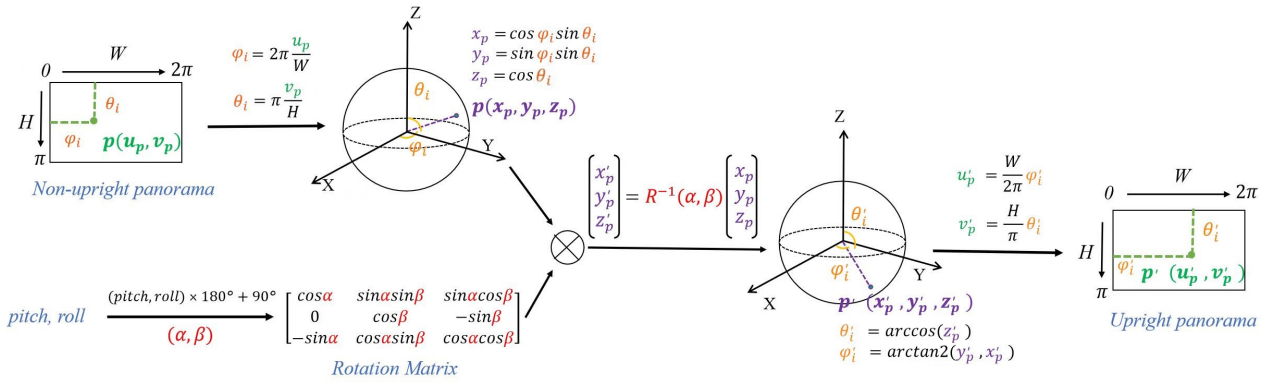


Figure 4: The process of spherical-model-based upright adjustment of the non-upright panorama with the estimated inclination angles.

non-upright 2D equirectangular image onto the spherical surface of the 3D unit-spherical-model centered on the camera. Then, pixels' positions on the spherical surface of the 3D unit-spherical-model are the imaging positions of scene around the camera when the camera is not upright. Next, according to the pitch angle and roll angle estimated by the network, we rotate the camera in the center of the spherical model from the non-upright state to the upright state, and the pixels' positions on the sphere surface also rotate with the camera rotation. Finally, projecting pixels on the sphere back to the equirectangular image, and the upright adjusted panorama is obtained.

To describe mathematically, suppose that the coordinate of a point in the non-upright 2D equirectangular image with the size of W×H is $p(u_p, v_p)$, then its spherical polar coordinates in the 3D unit-spherical-model is $(\varphi_i, \theta_i) = (2\pi\frac{u_p}{W},\ \pi\frac{v_p}{H})$, and its 3D coordinates in the camera coordinate system is $(x_p, y_p, z_p) = (cos\varphi_i \cdot sin\theta_i,\ sin\varphi_i \cdot sin\theta_i,\ cos\theta_i)$. If the normalized inclination angles of the panoramic camera estimated by the network are ***pitch*** and ***roll***, we first de-normalize these two angles to get α and β, then we can calculate the rotation matrix used to correct the camera from non-upright to upright, that is

$$\boldsymbol{R(\alpha,\beta)} = \begin{bmatrix} \boldsymbol{cos\alpha} & \boldsymbol{sin\alpha \cdot sin\beta} & \boldsymbol{sin\alpha \cdot cos\beta} \\ \boldsymbol{0} & \boldsymbol{cos\beta} & \boldsymbol{-sin\beta} \\ \boldsymbol{-sin\alpha} & \boldsymbol{cos\alpha \cdot sin\beta} & \boldsymbol{cos\alpha \cdot cos\beta} \end{bmatrix} \quad (4)$$

$R(\alpha,\beta)$ is the rotation matrix of the camera from the upright state to the current inclined state. For the 3D coordinates of all pixels on the sphere in the current camera coordinate system, we can obtain their 3D coordinates on the sphere in the upright camera coordinate system by multiplying the inverse matrix of $R(\alpha,\beta)$. Take 3D point $p(x_p, y_p, z_p)$ as an example, its 3D position $p'(x'_p, y'_p, z'_p)$ in the upright camera coordinate system can be calculated as follows,

$$\begin{bmatrix} \boldsymbol{x'_p} \\ \boldsymbol{y'_p} \\ \boldsymbol{z'_p} \end{bmatrix} = \boldsymbol{R^{-1}(\alpha,\beta)} \cdot \begin{bmatrix} \boldsymbol{x_p} \\ \boldsymbol{y_p} \\ \boldsymbol{z_p} \end{bmatrix} \quad (5)$$

In order to get the position of p' in the 2D panorama, we perform the inverse process of spherical model projection. We first calculate the polar coordinates of point $p'(x'_p, y'_p, z'_p)$ in upright spherical camera coordinate system: $(\varphi'_i, \theta'_i) = (arctan2(y'_p, x'_p),\ \arccos(z'_p))$. Then, the position of point p' in the 2D panorama is $(u'_p, v'_p) = (\frac{W}{2\pi}\varphi'_i,\ \frac{H}{\pi}\theta'_i)$. Perform the same operation as point p for all pixels in the non-upright image, and the upright adjusted image is obtained.

We can see that the calculation process of getting the upright adjusted image is very tedious. In order to accelerate this process, an accelerated vectorization algorithm was written for processing by the GPU, which can achieve real-time processing.

The final joint loss function is defined as: $\boldsymbol{L_{ang} + L_{img}}$.

4. Experiments

4.1. Experimental Setup

Data Generation: Like most existing methods, we use the SUN360 dataset [25], which contains a large variety of indoor and outdoor scenes. For all the images, pitch and roll angle values were randomly selected in the range ±90° to generate 67,000 non-upright labelled equirectangular images. Then, a 70-15-15% split was used for training, validation, and testing.

Training details: All models were trained for 300 epochs on a TITAN RTX 24Gb, using the Adam optimizer. The batch size was 16 and the learning rate was set to 1×10^{-4}.

4.2. Comparison with State-of-the-art Methods

We compared with several most recent panoramic camera inclination angles estimation methods, namely Cos2Fine [14], De360Up [11], vp-gscnn [15], LUTgan [16]. These methods employ more complex training

strategies, or design more complex network structures, or use representations with less distortion but require complex preprocessing as inputs. Due to Cos2Fine [14] and vp-gscnn [15] carry out the panoramic camera inclination angle estimation task in the range of $\pm 60^{\circ}$, to be fair, we evaluate the accuracy of the all methods within this range. The comparison results are listed in Table 1, and it can see that although the proposed method uses a concise backbone network consisting of only five cascaded down-sampling layers, and uses the most common equirectangular image as input, our method consistently outperforms these state-of-the-art methods across all tolerable error ranges due to the fusion of multi-scale shallow geometric feature attention mechanism that enhance the network's attention to the geometric deformation characteristics caused by the camera inclination, and pixel-level image loss that improves the convergence accuracy of the network. The importance of each module will be compared in detail in the ablation analysis in Section 4.3.

A latest work [30] published in WACV2024 was somewhat similar to the task in this paper, their method learns the projection relationship between non-upright images and upright images and directly generates upright panoramic images without estimating camera inclination angles. They used a relatively simple M3D dataset [24], which is a pure indoor scene dataset, unlike all the methods listed in Table 1, which are based on the more complex SUN360 dataset. For comparison with other methods, the author proposed a rough average error angle calculation algorithm, and compared with vp-gscnn [15] and De360up [11] within the tolerable error range of 5°: De360Up is **1.977°**, vp-gscnn is **1.807°**, and their method is **1.825°**. We used the same algorithm as [30] to calculate the average error of our method on M3D dataset, the value of the average error angle of our method is **0.848°**. The result proves that our method achieved highly competitive result, and we also list the accuracy of our method in different tolerable error ranges on M3D datasets in Table 2. Since the indoor scene contains more geometric information, it can be seen that our method has a more accurate angle estimation in M3D dataset than in SUN360 dataset under strict accuracy thresholds. In addition, it is worth noting that although the end-to-end method in [30] can directly generate upright panoramic images, it is easy to be limited by the image resolution, and the pixel distribution of the upright panoramic image generated by the network is inevitably biased from that of the true upright panoramic image. As mentioned before, in our method, the camera's attitude or the image of any resolution can be promptly corrected to upright attitude by utilizing the estimated angles from our network.

Methods	Accuracy in tolerable error ranges (%)					
	1°	2°	3°	4°	5°	12°
Cos2Fine (2019)	30.1	51.7	65.9	74.0	79.1	91.0
De360Up (2019)	7.1	24.5	43.9	60.7	74.2	97.9
vp-gscnn (2020)	19.7	53.6	75.5	87.2	92.6	98.4
LUTgan (2023)	29.9	65.3	80.3	86.3	89.2	95.2
Ours	**31.2**	**72.9**	**89.8**	**95.5**	**97.5**	**99.5**

*The data of other methods is referred to vp-gscnn [15] or data listed in their paper.

Table 1: Comparison Results on SUN360 Dataset ($\pm 60^{\circ}$)

4.3. Analysis of Our Methods

Ablation analysis: Subsequently, an ablation analysis was performed using the public test set of Shan et al. [14] to verify the generalization performance of our method and the importance of each module. In Table 3, five variations of the proposed network were generated through different combinations of its main components: the image loss module, the attention module, and the ReLU module in attention feature fusion. The last line shows the performance when all modules are included, and it is evident that the accuracies are obviously higher than other combinations. Comparing the last two rows of Table 3, it can also be seen that when fusing the attention features corresponding to formula 1, compared with simply adding the attention feature maps and backbone network feature maps, it is crucial to add ReLU nonlinear activation module

Method	Accuracy in tolerable error ranges (%)					
	1°	2°	3°	4°	5°	12°
Ours	70.3	89.2	93.0	94.5	95.5	97.4

Table 2: Performance of Our Method on M3D Dataset ($\pm 90^{\circ}$)

	Image Loss	Attention	ReLU	Accuracy in tolerable error ranges (%)						
				1°	2°	3°	4°	5°	10°	12°
Backbone Network	×	×	×	17.8	53.3	72.4	84.9	90.9	97.7	98.1
	√	×	×	19.4	53.6	71.9	85.7	91.5	97.8	98.5
	×	√	×	21.9	58.5	74.8	87.0	93.5	99.2	99.2
	√	√	×	23.6	61.1	79.6	91.1	95.3	98.9	99.1
	√	√	√	**30.9**	**74.6**	**90.4**	**96.0**	**97.8**	**99.3**	**99.5**

Table 3: Ablation study for different combinations of independent components on the public test set of Shan et al. [14]

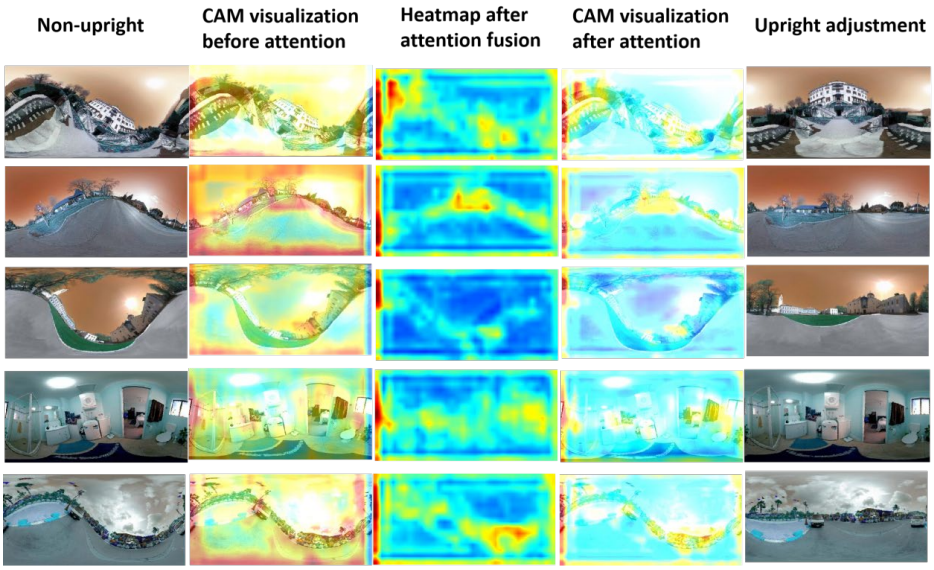


Figure 5: Visualization of attention mechanism. The attention causes the network to focus on key features. The heat map after attention fusion shows a high geometric accordance with the input.

Datasets from SUN360 [25]	Accuracy in tolerable error ranges (%)											
	1°	2°	3°	4°	5°	6°	7°	8°	9°	10°	11°	12°
±30°	35.3	77.0	91.5	95.8	97.6	98.5	98.9	99.1	99.2	99.2	99.3	99.3
±60°	31.2	72.9	89.8	95.5	97.5	98.3	98.8	99.1	99.2	99.3	99.4	99.5
±90°	26.9	62.1	77.8	84.6	88.0	90.1	91.4	92.3	93.0	93.7	94.2	94.6

Table 4: The performance of our method in the range of different camera inclinations.

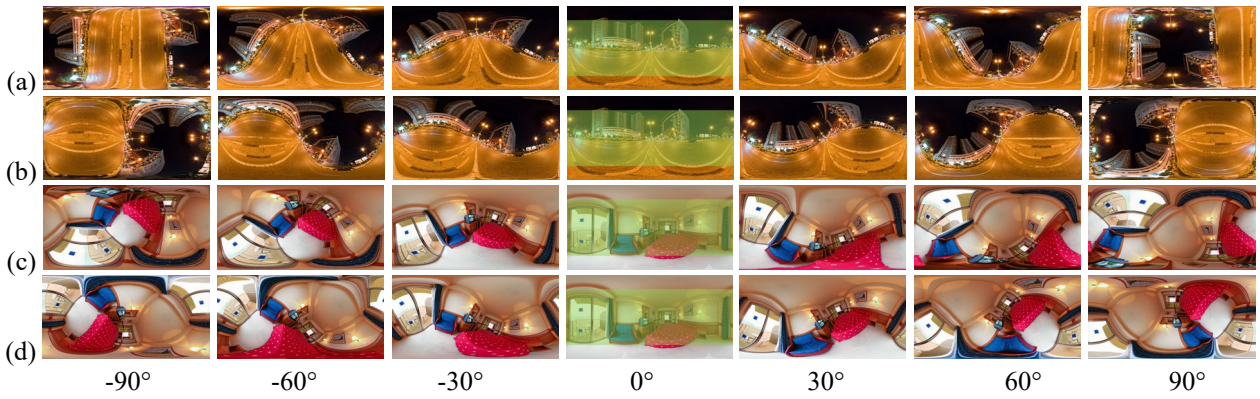


Figure 6: Influence of the pitch and roll angles of camera on the equirectangular image. From left to right, (a) and (c): roll = 0°, pitch = -90°, -60°, -30°, 0°, 30°, 60° and 90° respectively. (b) and (d): pitch = 0, roll = -90°, -60°, -30°, 0°, 30°, 60° and 90° respectively. For the middle column, the regions marked with light green shadow in these equirectangular images are the regions with small distortion and richest geometric information.

on this basis. The nonlinear activation function ReLU can increase the sparsity of the feature maps and the nonlinearity of the network, so that the network has stronger learning ability and achieves higher prediction accuracy.

Visualization of attention mechanism: Fig. 5 shows several example images where the class activation map (CAM) was obtained before and after adding the attention module, and the heatmap of each image after attention fusion is also shown. Through a comparison of images in

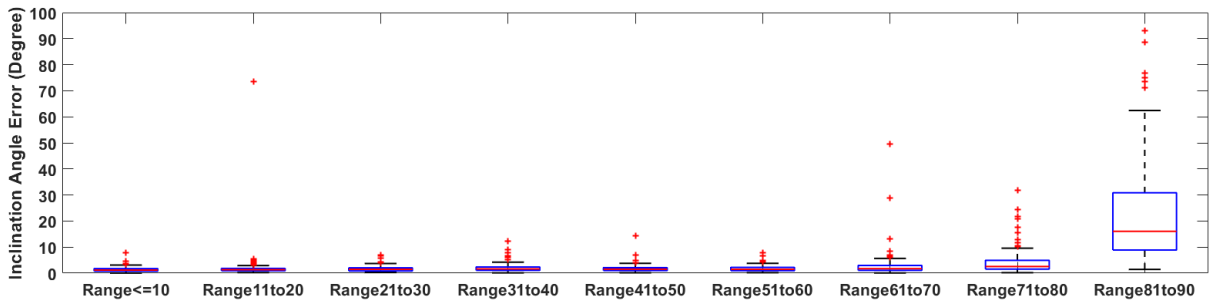


Figure 7: The summarizing box plots for the angle error distribution over different camera inclination ranges between the estimated camera inclination angle of our method and the true value.

column 4 and column 2, it is evident that attention fusion allows the network to focus on the key features. This advantage is more apparent in the heatmap diagram of the third column. We can see that the heatmap of each image after attention fusion shows a high geometric consistency with the input. During the learning process, the network with attention fusion will pay more attention to the distortion of vertical and horizontal lines in the image, such as horizon lines or the horizontal and vertical outlines of various objects. This attention to geometric information is similar to the visual processing performed by humans when correcting a non-upright image.

Limitations of our method: We discuss the performance of our method in the range of different camera inclinations: Including the range of $\pm 30^{\circ}$, the range of $\pm 60^{\circ}$, and the range of $\pm 90^{\circ}$. The data came from the test set of SUN360, which contains a total of 10,108 equirectangular images with different camera inclination angle in the range of $\pm 90^{\circ}$, including 4,534 images in the range of $\pm 60^{\circ}$ and 1,155 images in the range of $\pm 30^{\circ}$. The pretrained model used for evaluation was trained on the train set of SUN360 in the range of $\pm 90^{\circ}$.

The results are shown in Table 4, it can be seen that our method always performs well when the camera's pitch and roll angles in the range of $\pm 60^{\circ}$. However, when the range of camera's pitch and roll angles is extended to $\pm 90^{\circ}$, the performance of our method has a relatively significant decline in accuracies. One possible reason is the influence of the representation of panoramic images used for training. As shown in Fig. 6, when the absolute values of the pitch angle and the roll angle of the camera are closer to 90°, the distortion of the region with rich geometric information in the upright equirectangular image becomes more serious, and some original horizontal lines in upright images are closer to vertical lines, and some original vertical lines are closer to horizontal lines.

In order to further analyze this phenomenon, we drew summarizing box plots for the angle error distribution over different camera inclination ranges between the estimated camera inclination angle of our method and the true value, which are shown in Fig. 7. Specifically, for the test set of SUN360, according to the absolute value of the pitch and roll angles of each non-upright image, we first screened out 9 groups of images in the range of 0 to 90 degrees with an interval of 10 degrees, then the prediction angle error distribution of each group of images is analyzed. For each box in Fig. 7, the central red line indicates the average error, and the bottom and top edges of the box indicate the 25th and 75th percentiles (q1 and q3) of all error values in the corresponding group, respectively. The dashed line extends to the most extreme error value that are not considered outliers, while the outliers are plotted separately using the "+" symbol. Error values are considered outliers if greater than $q3 + w \times (q3-q1)$ or less than $q1 - w \times (q3 - q1)$, where w is the maximum length of the dotted line [14]. It can be seen that similar to the phenomenon in Table 4, the performance of our method does have a significant decline when the inclination angle of pitch and roll is close to 90 degrees, but our method consistently performs well until the inclination angle of pitch and roll approached 80 degrees.

Using other representations of panoramic image such as discrete spherical image to train the network may be able to better cope with the problem of inclination angles estimation under large camera inclination. However, a camera is usually not tilted beyond the range of $\pm 80^{\circ}$ in practice, and the accuracies of our method are still acceptable even if the range of tilt angle is $\pm 90^{\circ}$.

5. Conclusions

In this paper, a multi-scale attention-based network is proposed to achieve inclination angles estimation for panoramic camera. Both local distortion features and global shallow geometric features at two scales are used to guide the training of the network. To the best of our knowledge, this is the first time a pixel-wise image loss is introduced into the panoramic camera rotation angle estimation. A quantitative and qualitative evaluation and ablation analysis confirmed the validity of the proposed method, and the results of the comparison experiment also demonstrated that the proposed multi-scale attention-based inclination angles estimation method for panoramic camera outperforms the state-of-the-art methods.

References

[1] Coors, Benjamin, Alexandru Paul Condurache, and Andreas Geiger. "Spherenet: Learning spherical representations for detection and classification in omnidirectional images." Proceedings of the European conference on computer vision (ECCV). 2018.

[2] Jin, Lei, et al. "Geometric structure based and regularized depth estimation from 360 indoor imagery." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[3] He, Yuhang, et al. "Know your surroundings: Panoramic multi-object tracking by multimodality collaboration." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[4] Cowley, Anthony, Ian D. Miller, and Camillo Jose Taylor. "UPSLAM: Union of panoramas SLAM." 2021 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2021.

[5] Sun, Cheng, Min Sun, and Hwann-Tzong Chen. "Hohonet: 360 indoor holistic understanding with latent horizontal features." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[6] Chiyu" Max" Jiang, et al. "Spherical CNNs on Unstructured Grids." ICLR (Poster). 2019.

[7] Jung, Jinwoong, et al. "Upright adjustment of 360 spherical panoramas." 2017 IEEE Virtual Reality (VR). IEEE, 2017.

[8] Goto, Tsubasa, et al. "Line-based global localization of a spherical camera in manhattan worlds." 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018.

[9] Bazin, Jean-Charles, et al. "Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment." The International Journal of Robotics Research 31.1: 63-81, 2012.

[10] Jung, Jinwoong, et al. "Robust upright adjustment of 360 spherical panoramas." The Visual Computer 33: 737-747, 2017.

[11] Jung, Raehyuk, et al. "Deep360Up: A deep learning-based approach for automatic VR image upright adjustment." 2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR). IEEE, 2019.

[12] Jung, Raehyuk, Sungmin Cho, and Junseok Kwon. "Upright adjustment with graph convolutional networks." 2020 IEEE International Conference on Image Processing (ICIP). IEEE, 2020.

[13] Jeon, Junho, Jinwoong Jung, and Seungyong Lee. "Deep upright adjustment of 360 panoramas using multiple roll estimations." 14th Asian Conference on Computer Vision, (ACCV). Springer International Publishing, 2019.

[14] Shan, Yuhao, and Shigang Li. "Discrete spherical image representation for cnn-based inclination estimation." IEEE Access 8: 2008-2022, 2019.

[15] Davidson, Benjamin, Mohsan S. Alvi, and João F. Henriques. "360 camera alignment via segmentation." European Conference on Computer Vision. Cham: Springer International Publishing, 2020.

[16] Chen, Heyu, Shigang Li, and Jianfeng Li. "An End-to-End Network for Upright Adjustment of Panoramic Images." Procedia Computer Science 222: 435-447, 2023.

[17] Rey-Area, Manuel, Mingze Yuan, and Christian Richardt. "360monodepth: High-resolution 360deg monocular depth estimation." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.

[18] Zhuang, Chuanqing, et al. "ACDNet: Adaptively combined dilated convolution for monocular panorama depth estimation." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 36. No. 3. 2022.

[19] Weyn, Jonathan A., Dale R. Durran, and Rich Caruana. "Improving data-driven global weather prediction using deep convolutional neural networks on a cubed sphere." Journal of Advances in Modeling Earth Systems 12.9: e2020MS002109, 2020.

[20] Rashed, Hazem, et al. "Generalized object detection on fisheye cameras for autonomous driving: Dataset, representations and baseline." Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2021.

[21] Gao, Shaohua, et al. "Review on panoramic imaging and its applications in scene understanding." IEEE Transactions on Instrumentation and Measurement 71: 1-34, 2022.

[22] Delmas, Sarah, et al. "SpheriCol: A driving assistant for power wheelchairs based on spherical vision." IEEE Transactions on Medical Robotics and Bionics, 2023.

[23] Su, Yu-Chuan, and Kristen Grauman. "Kernel transformer networks for compact spherical convolution." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019.

[24] Chang, Angel, et al. "Matterport3D: Learning from RGB-D Data in Indoor Environments." 2017 International Conference on 3D Vision (3DV). IEEE Computer Society, 2017.

[25] Xiao, Jianxiong, et al. "Recognizing scene viewpoint using panoramic place representation." 2012 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2012.

[26] Demonceaux, Cédric, Pascal Vasseur, and Claude Pégard. "Robust attitude estimation with catadioptric vision." 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2006.

[27] Demonceaux, Cédric, Pascal Vasseur, and Claude Pégard. "Omnidirectional vision on UAV for attitude computation." Proceedings 2006 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2006.

[28] Bazin, Jean-Charles, et al. "Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment." The International Journal of Robotics Research 31.1: 63-81, 2012.

[29] Takikawa, Towaki, et al. "Gated-scnn: Gated shape cnns for semantic segmentation." Proceedings of the IEEE/CVF international conference on computer vision, 2019.

[30] Liu, Jingguo, et al. "Generation of Upright Panoramic Image From Non-Upright Panoramic Image." Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2024.