SkyCloud: Neural Network-based Sky and Cloud Segmentation from Natural Images

1st Christoph Gerhardt
Ilmenau University of Technology
Ilmenau, Germany
christoph.gerhardt@tu-ilmenau.de

2nd Florian Weidner
Ilmenau University of Technology
Ilmenau, Germany
florian.weidner@tu-ilmenau.de

3rd Wolfgang Broll
Ilmenau University of Technology
Ilmenau, Germany
wolfgang.broll@tu-ilmenau.de

***Abstract*—The comprehensive understanding of outdoor scenes is a necessary requirement for a wide variety of applications. For example, semantic segmentation enables applications such as outdoor robot navigation, image stylization, weather forecasting, or climate monitoring. However, existing outdoor scene understanding models are often less reliable in challenging situations such as changing weather conditions or low light. Additionally, current approaches mainly focus on sky and ground separation and do not incorporate valuable information provided by weather conditions and cloud coverage. To overcome these challenges, we present *SkyCloudNet*, a multi-task neural network architecture that extracts high-level attributes from the input image and utilizes them to improve the robustness of the network to environmental influences. Furthermore, it allows for the segmentation of cloud segments in natural outdoor images. While existing cloud segmentation approaches are limited to cropped sky-only images, our model enables the segmentation from entire landscape images with arbitrary resolution. Next to that, *SkyCloudNet* achieves state-of-the-art performance in environmental attribute estimation and sky segmentation. As cloud segmentation from natural images has not been addressed in previous literature, we also release the *SkyCloud* data set consisting of 350 high-resolution outdoor images with dense labels of sky and cloud segments.**

***Index Terms*—Semantic segmentation, scene understanding, image analysis, neural networks, data sets**

I. Introduction

In a wide variety of application domains, researchers and practitioners alike take advantage of data extracted from outdoor images. For example, the sky location in an image is essential for image stylization [1], outdoor robot navigation [2], and automatic sky replacement in augmented reality [3]. Information on clouds and cloud coverage is used for solar energy measurements and predictions [4], weather forecasting [5], and climate change monitoring [6], [7]. In all these use cases, having detailed and expressive data is crucial — without it, a holistic and comprehensive understanding of the situation is impossible. For example, short-term weather forecasting models require large amounts of dynamic data to form accurate predictions [8], [9]. However, most current cloud data collection systems are based on ground-based wide-angle cameras [10], [11] or satellite images [12], [13]. While such images can comprehensively capture a large sky area, the restriction to specific hardware limits scalability (i.e., wide-angle cameras) or the data has a rather low-resolution (i.e., satellites).

The possibility of extracting sky and cloud information from arbitrary outdoor images would result in an enormous amount of additional data (e.g., from stationary webcams or geo-tagged social media posts) that could support the mentioned application domains. To enable this, we present *SkyCloudNet*, a multi-task network architecture which allows for the segmentation of sky and cloud pixels in arbitrary outdoor images. Our approach eliminates the need for cropped sky-only images that current cloud segmentation approaches require. Furthermore, as existing approaches for sky segmentation often work well under ideal conditions (i.e., daylight, blue sky) but fail under more challenging circumstances such as low-light or rainy conditions[14], [3], [15], we extract environmental attributes (i.e., season, time of day and weather) from the input image and incorporate this information into the segmentation process of our network. Together, we present the following core contributions:

- A neural network architecture for sky segmentation that facilitates high-level attribute estimation for robust segmentation across various environmental conditions.
- A semi-supervised learning pipeline for the segmentation of clouds in natural images.
- A benchmark data set for cloud segmentation from natural images.

The *SkyCloud* data set, the *PyTorch* implementation of *SkyCloudNet*, pre-trained weights, and the code to reproduce the training and testing data sets are publicly available.[1]

II. Related work

A. High-level Outdoor Attribute Estimation

Image attributes are high-level descriptions of a visual property that provide semantic context. With regard to outdoor images, such attributes usually correspond to environmental factors such as time, seasonal, or weather conditions, but they could also refer to the location of the image, or other properties of the scene.

An early work in this field is the *Archive of Many Outdoor Scene* (AMOS) data set and the corresponding analysis [16].

2023 IEEE. This is the author's version of the article that has been published in the proceedings of 8th International Conference on Image, Vision and Computing (ICIVC) conference. The final version of this record is available at https://doi.org/10.1109/ICIVC58118.2023.10270450

[1] https://github.com/carhartt21/SkyCloudNet

It includes millions of images from static webcams captured over a long period of time. The authors used the data set to investigate the variations in the scene based on weather conditions, human activity, and changes in season [17]. They also augmented the data set with real weather information from nearby weather stations [18]. However, their user tests showed that real data does not necessarily match the scene attributes as perceived by humans — pointing out the ambiguity during scene understanding in outdoor images (a problem we also encounter and discuss in Section VII).

Laffont et al. [19] used regressors to estimate 40 scene attributes, including lighting, weather, seasons, and subjective impressions from images. The corresponding data set contains 8,571 images from 101 webcams with 40 attribute labels derived from a crowd-source experiment. This data set was used by Baltenberger et al. [20] to train a deep neural network with a separate classification layer for each attribute. Their results show minimal improvements in classification accuracy (-0.4% in error rate), but a significant gain in computational performance (18× faster). Lu et al. [21] proposed a collaborative learning framework to assign the attributes *sunny* or *cloudy* to images. To do this, they used a combination of five weather features, i.e., sky, shadow, reflection, contrast, and haze, calculated from different feature extractors. Similarly, Chu et al. [22] proposed a framework that combines real weather information and location with images from *Flickr*. Their final data set consists of approximately 119,000 images that cover five attribute classes (sunny, cloudy, snowy, rainy, and foggy).

None of the mentioned approaches for attribute estimation are well suited for general use cases, as they suffer from limited scene variation [19] or a low number of attribute classes [16], [21], [22]. We alleviate these issues by combining multiple data sets during the training process of our network.

B. Sky and Cloud Segmentation

Although **sky segmentation** is frequently used as an auxiliary resource in different applications [1], [21], [23], [24], [25], [26], it was rarely a primary research focus. The introduction of the *Skyfinder* data set by Mihail et al. [14] composed of 100,000 images from 53 stationary webcams changed that and provided a reference point for further developments. The authors used the data set to compare the results of three existing sky segmentation approaches [24], [23], [21]. The results showed a general need for improvements and highlighted the influence of lighting and weather conditions on segmentation accuracy. Building upon this data set, La Place et al. [27] used *RefineNet* [28] to improve segmentation accuracy (84.05% mIoU). To increase flexibility, Nice et al. [2] presented an adaptive approach that automatically selects a suitable candidate from 13 conventional image processing techniques (e.g., Sobel filters, shift segmentation), based on the input image. However, the reported overall accuracy is still relatively low (82% accuracy), limiting its use in real-life scenarios.

Although sky segmentation can be addressed through general-purpose semantic segmentation networks [29], [30], [31], these approaches do not explicitly consider the effects of environmental changes. To address this issue, several strategies to improve outdoor semantic segmentation results under adverse conditions have been presented. These include augmentation with artificial data [32], [33], [34], pre-processing [35], [33], [36], and data fusion (either with additional sensor data [37], [38] or images [39], [40]). Artificial data augmentation and pre-processing are viable options if the number of adverse conditions is limited (e.g., fog [32], rain [35], night [33]), but not applicable in more diverse and complex scenarios (e.g., different seasons). Data fusion is only applicable in setups that actually have several data sources (e.g., robots or automated driving), which is not the case in our scenario. To the best of our knowledge, only Liba et al. [15] investigated sky segmentation with regard to environmental conditions. They presented a post-processing technique that uses alpha-masks for edge refinement in low-light situations. In addition to lighting changes, our approach also works for other environmental conditions (i.e., season and weather).

In comparison to sky classification, which only separates sky and non-sky regions, **cloud segmentation** further separates the sky region into clear sky and cloudy segments. Naïve methods for cloud segmentation extract color features from sky-only images and classify cloud and sky regions using fixed thresholds [41], [42]. Such methods only work under specific conditions but lack the flexibility to adapt to different environmental conditions and types of clouds [43]. To increase flexibility, Liu et al. [44] divide the image into a series of irregular segments and then calculate a local threshold for each segment. Similarly, Li et al. [45] use a combination of minimum cross-entropy [46] and fixed thresholds. Dev et al. [47] present a *U-Net*-based approach for cloud segmentation. Like all the previously mentioned approaches, their network requires sky-only images as input.

With regard to data sets, the *HYTA* data set [45] consists of 32 sky-only images showing sky/cloud scenarios under varying illumination conditions with binary segmentation maps. The labels were later extended to include ternary segmentation maps, which label thin and thick clouds individually [48]. Similar data sets with low-resolution sky-only images have been presented [49], [10], [11].

To the best of our knowledge, no previous approach addressed the segmentation of clouds in images that are not restricted to a cropped sky-only region. Therefore, we present a new multi-task approach that combines sky segmentation (cf. Section IV-B) and cloud segmentation (cf. Section IV-C) into a single framework. In addition, we use high-level attribute features (cf. Section IV-A) to increase the robustness to environmental changes. high-level attribute estimation. For evaluation purposes and as a reference for future work, we also present a new data set for cloud segmentation from natural images (cf. Section III-B). Due to the absence of an adequate training data set, we used a semi-supervised approach for the training of our cloud segmentation head.

TABLE I
COMPOSITION OF THE COMBINED MULTI-TASK DATA SET.

	#Images	#Sky labels	#Attribute labels
Transient Attributes [19]	8,673	8,673	8,673
SkyFinder [14]	98,687	98,687	98,687
OUTSIDE15k [50]	15,000	15,000	-
Flickr	32,559	-	32,559
Image2Weather [22]	1,729	-	1,729
Total	156,648	122,360	141,648

III. DATA SETS

A. Attribute Estimation and Sky Segmentation

Insufficient scene variation and limited generalizability are the two main issues we identified when reviewing existing attribute estimation data sets in Section II. This section describes the data set we used to train our multi-task network presented in Section IV. For this, we require consistent labeling and a sufficient number of images per class in each task. No existing data set met these criteria. Thus, we combined parts of existing data sets. We started with the 40 "transient" attributes provided by Laffont et al. [19] and removed subjective classes (e.g., mysterious, soothing). Then we combined classes that showed high cross-correlation (e.g., snow and ice, dry and warm) and grouped the remaining classes into the following categories:

- **Time of day:** daytime, sunset/sunrise, dusk/dawn, night
- **Season:** spring, summer, autumn, winter
- **Weather:** sunny, snow, rain, fog

The resulting data set of approx. 8,700 images was not well balanced, as the classes *autumn*, *spring*, *rain*, and *fog* were heavily underrepresented (present in < 0.1 % of the images). To mitigate this issue, we added approx. 1,700 images with the label *fog* and *rain* from the *Image2Weather* data set [22]. Furthermore, we complement the data set with approx. 32,500 creative common licensed images from *Flickr*. Here, we selected the images based on image tags with the aim of strengthening the underrepresented attribute classes. [2] Furthermore, we added the *SkyFinder* data set [14] (approx. 100,000 images) as well as the *OUTSIDE15k* data set [50] (15,000 images). These images contain a sky segmentation mask but, by default, no attribute labels. However, attribute probabilities calculated using the regressors from Laffont et al. [19] are available for the *SkyFinder* data set. Since both data sets have very similar characteristics (stationary webcams with low resolution), we found those attributes to be accurate enough to include them in our training. To do so, we converted the probabilities to class labels using a confidence threshold of 80% (as proposed by Laffont et al.) to find strong positive samples. In addition, we manually created sky segmentation masks for the 101 scenes in the *Transient Attribute* data set and obtained approx. 8,600 additional sky segmentation labels.

The number of images and labels in our final training data set for outdoor attribute estimation and sky segmentation is shown in Table I. The large number of images and scene variations in our combined data set improve the generalizability and segmentation accuracy of the trained model (cf. Section VI).

[2]The search terms, the list of the images, and the extracted attributes can be found in the corresponding repository.

B. Cloud Segmentation

Due to the absence of an adequate data set for cloud segmentation from natural images, we created a new data set. The *SkyCloud* data set contains 350 images captured at different locations, seasons, and times of day. For annotation, we follow a ternary class approach, with the labels *sky*, *thin clouds*, and *thick clouds* as used in the *HYTA* data set [48]. The label *thin cloud* corresponds to high-altitude clouds mainly composed of ice crystals (e.g., cirrus, cirrocumulus or cirrostratus), whereas the label *thick cloud* refers to clouds formed by water droplets in low and medium heights (e.g., altocumulus, stratocumulus or stratus). This distinction is especially relevant for use cases like climate monitoring or solar energy measurements. However, the manual assignment is not always trivial. Therefore, we derived general labeling guidelines: We label segments as *thick cloud* if the sky behind it is not visible. The label *thin cloud* is assigned if the following conditions are met: (i) the sky needs to be clearly visible through the cloud, and (ii) the cloud needs to have a visible structure that separates itself from the rest of the sky. This implies that haze and smog do not belong to any cloud class but should be rendered as the sky.

To increase the level of detail and the labeling consistency across the data set, we used a two-round labeling approach. In the first round, the images were assigned to different annotators. In the second round, a single annotator refined these labels and reviewed them for consistency.

The final data set consists of 350 images with dense labels of sky, thin clouds, and thick clouds. To the best of our knowledge, the *SkyCloud* data set is the most extensive data set with cloud labels on ground-based images and the only one that is not limited to sky-only images. However, it is essential to note that the data set is primarily intended for evaluation purposes, as the number of images is not well suited for traditional neural network training. Consequently, we do not use the data set in our training stage, but only in our analysis in Section V and hope that our results can be used as a reference for future developments in this research area.

IV. NEURAL NETWORK ARCHITECTURE

In this section, we will discuss *SkyCloudNet*, our proposed multi-task network architecture. Fig. 1 provides an overview. The network follows a cascading principle, where the output of one task is used as input for the next. For the main evaluation in Section V, we use *ResNet-50* as the backbone. We present the results of alternative architectures in the ablation study in Section VI. In the following, we discuss the single components shown in Fig. 1.

A. Attribute Estimation Head

For the attribute estimation head, we decided to use a structure that follows the structure of previous image classification tasks [51]. It consists of three parallel branches, whereas

each branch predicts the attributes for one of the categories mentioned in Section III (time of day, weather, season). Each branch consists of a series of three strided 3×3 convolutions, followed by an average pooling layer and a linear classifier to form the final class prediction. We then calculate L_1 for each category and determine the total attribute estimation loss as an average over the three categories (*time of day*, *weather*, *season*).

B. Sky Segmentation Head

The structure of the sky segmentation head is displayed in Fig. 2. In the beginning, we concatenate the feature map from the attribute estimation with the result of the backbone. Next, we employ parallel 2-dimensional adaptive average pooling layers with bin sizes of: 1×1, 2×2, and 4×4. This has been shown to increase the network's flexibility to varying input resolutions, improve the robustness to object variation and changes to the spatial layout [52], and expands the effective receptive field of the network [53]. To combine the results of the individual pooling sizes, we utilize a convolutional layer after each pooling layer and concatenate the resulting feature maps. Afterward, we pass them through a 1×1 convolution.

Next, we merge the resulting features with intermediate results from the backbone network (based on *Feature Pyramid Networks* [54]). To combine the feature maps, we first align them and then perform an element-wise summation. Alignment is performed by upsampling the feature maps using bilinear interpolation along with a 1×1 convolution.

In the final step, we concatenate all feature maps — again using bilinear interpolation — to align their sizes. This step combines the semantically rich features of pooling with the spatially more robust feature maps from the backbone network. To obtain the final segmentation results, the feature maps are passed to a final convolution, followed by softmax activation for class calculation. The segmentation results are then resized to the original input size.

During training, we use the segmentation results to calculate an accuracy loss λ_{acc} between the prediction and the ground truth using *cross entropy* (CE):

$$\lambda_{acc} = CE(pred, gt) \tag{1}$$

To further adapt our network for the specific task of sky segmentation and to account for the fact that not all images in our training data set include sky segmentation labels, we add an auxiliary region loss λ_{reg}. This loss is based on the assumption that sky and non-sky regions in an image are usually large continuous segments. Therefore, we intend to minimize the number of individual regions in the final segmentation. To do this, we add a function r_c that counts the regions in the segmentation. We determine the number of regions in the prediction and the ground truth and calculate a L_1-loss with:

$$\lambda_{reg} = \frac{||r_c(pred) - r_c(gt)||_1}{r_c(pred)} \tag{2}$$

If ground truth data is unavailable, the network should aim for segmentation into one non-sky and one sky region and therefore set $r_c(gt)$ to 2. Although this assumption obviously is not true for all image layouts, we still found it to significantly reduce the amount of small misclassified regions and improve the overall accuracy (cf. Section V).

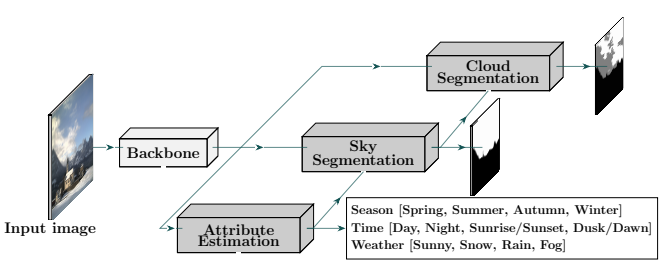

Fig. 1. *General structure of our network.*

C. Cloud Segmentation Head

The structure of the cloud segmentation head is similar to the sky segmentation head. Instead of the attribute feature maps from the attribute estimation head, we use the result of the sky segmentation as additional input. Due to the absence of a labeled data set, we use a semi-supervised training process (see Fig. 3). In the lower left branch, we pass an image I through the backbone network (BN) to get a spatially dense feature map. We then apply a binary mask derived from the sky detection head to those features. Next, the cloud segmentation head h_c produces a feature representation for every sky pixel p of the input image. To calculate pseudo-labels, we perform k-means clustering with pre-calculated centroids that are periodically updated during the training process. The goal of clustering is to minimize the distance between the labels from the cloud segmentation head and the pseudo-labels. Therefore, it corresponds to the following objective:

$$\min \sum_i ||h_c(BN(I))[p] - \mu_{c_{ip}}||^2 \tag{3}$$

where c_{ip} is the cluster label of pixel p in image i, and μ the centroid of the corresponding cluster. The initial centroids are determined in a pre-training step (cf. Section IV-D)

Learning feature representation from clustering will result in clusters that are compact in feature space but do not guarantee that the classification will give the desired semantic results. Therefore, additional training objectives must be added to the process. Thus, we create a second view of the input and use it to enforce spatial continuity and equivalence to geometric transformation by comparing the classification results of both views (cf. Ouali et al. [55] and Cho et al. [56]). To create the auxiliary view, we use the sky segmentation result (denoted by the dashed line on the lower left in Fig. 3) and randomly crop a part of the sky region. This cropped region is used as an input for the upper branch in Fig. 3. The auxiliary view is created by a geometric transformation G that includes cropping a random-sized rectangle from the sky region and random horizontal flipping. We pass this view through the network and calculate a second cluster loss.

Next, we calculate the cluster loss for each view as follows. Letting bn denote the backbone network, we represent

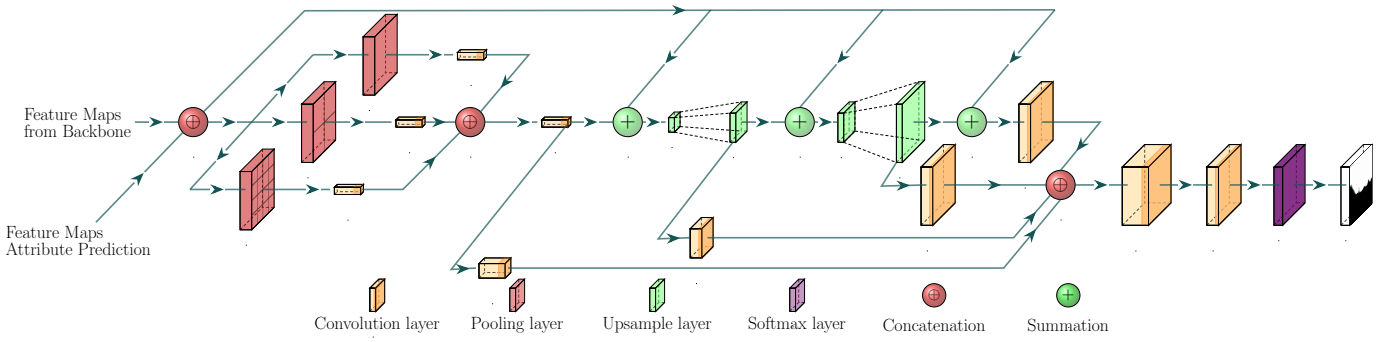

Fig. 2. *Structure of the sky segmentation head.*

the final feature map of the cloud segmentation head with $H_C(I) = h_c(bn(I))$ and the pixel-level feature representation with $H_C(I)[p]$, respectively. Based on the pseudo-labels we extract from the k-means clustering, we calculate a cross-entropy (CE) loss between the prediction and derive the pseudo-labels from the k-means clustering as:

$$\lambda_{clust} = -\log \frac{e^{-d_{\cos}(H_C(I)[p], \mu_{c_{ip}})}}{\sum_l e^{-d_{\cos}(H_C(I)[p], \mu_l)}} \quad , l \in \{1, \ldots, k\} \quad (4)$$

with $d_{\cos}$ denoting the cosine distance function and k the number of clusters.

Subsequently, we calculate the continuity loss λ_{cont} between the original and the auxiliary view using a L_2 loss function:

$$\lambda_{cont} = \sum_p (H_c(G(I))[p] - G(H_c(I))[p]))^2 \quad (5)$$

In contrast to previous approaches, we do not use the feature maps, but the final segmentation results as input for our loss function. This is possible because we have a known set of classes. The total loss for the cloud segmentation head is:

$$\lambda_{cloud} = w_{clust}(\lambda_{clust_1} + \lambda_{clust_2}) + w_{cont}\lambda_{cont} \quad (6)$$

where w_{clust} and w_{cont} are weights that we use for loss balancing.

D. Implementation Details

1) Pre-training: To use k-means clustering for pseudo-labeling in the cloud segmentation head, we pre-calculate the cluster centroids before training using the GPU mini-batch k-means algorithm [57]. The centroids were calculated with 20 batches with a batch size of 100. We used the ground truth sky segmentation maps to mask non-sky regions and ensure that only sky and cloud pixels are used for the clustering.

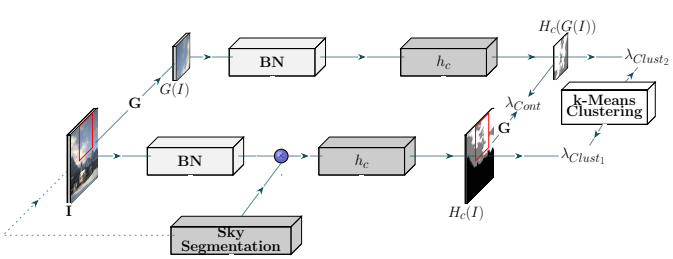

Fig. 3. *Visualization of the semi-supervised training process of the cloud segmentation head.*

To match the centroids to the classification classes, we pre-trained the classifier using sky-only images from the *HYTA* data set [45]. We applied data augmentation (cropping, mirroring, and Gaussian blur) and pre-trained the classification head for one epoch with 500 iterations.

2) Training: Before feeding an image to the network, we normalize the color space and transform the pixel values to a range of $[0, 1]$. Afterward, we calculate normalized pixel values for each color channel based on statistical values from *ImageNet* [58]. During training, the learning rate is periodically updated using the polynomial learning rate. We set the initial learning rate to 0.02, and the power to 0.9. Momentum and weight decay are set to 0.9 and 0.0001. For gradient optimization, we use *SGD* [59]. During the training, we randomly augmented the data using cropping, horizontal flip, and resizing.

Since the semi-supervised cloud segmentation head depends on accurate sky segmentation masks, we initially trained the backbone with only the attribute and the sky segmentation head for 20 epochs. After this initial training stage, we trained all components together for 20 additional epochs to acquire the final cloud and sky segmentation weights for the evaluation (cf. Section V). To achieve a fair comparison, we only use the weights from the first stage when evaluating our results for attribute and sky classification.

V. Evaluation

In this section, we will evaluate the performance of our multi-task network on each task using existing data sets and report the results in the combined task of sky and cloud segmentation using the presented data set (cf. Section III). To separate the existing data sets into train and test sets, we followed the instructions of the corresponding authors. When such instructions were not available, we used an 80/15/5 split for training, testing, and evaluation data sets.

A. Attribute Estimation

We first tested the performance of attribute estimation using the *Transient Attribute* data set [19]. As a reference, we used the regressors released with the data set and *TransientNet* [20]. For performance assessment, we calculate the overall accuracy

TABLE II
RESULTS ON ATTRIBUTE ESTIMATION ACCURACY [%]

	Laffont et al. [19]	*TransientNet* [20]	SkyCloudNet
Time	95.5	94.3	**99.1**
Season	92.3	92.4	**98.2**
Weather	93.2	92.8	**97.3**
Average	94.2	93.8	**98.2**

as the number of correct predictions divided by the number of all samples in the test set. We report the score for each category individually and as an averaged score over all classes in Table II. We achieve superior classification accuracy. The accuracy for the *"time of day"* category was especially high. This was expected, given that the corresponding classes were the most prominent classes in our training data set.

B. Sky Segmentation

Next, we wanted to know how our proposed network performed in the sky segmentation task. Here, we used pixel-wise accuracy and *intersection over union* (IoU) as performance measures. Pixel-wise accuracy represents the ratio of correctly classified pixels to the number of total pixels. *IoU* represents the ratio of the area of overlap and the area of union between the predicted segmentation labels and the ground truth. The *mean IoU* (mIou) states the average *IoU* over all classes.

We chose *ResNet-50* with a linear layer as a classifier as the baseline. Additionally, we tested a selection of popular state-of-the-art semantic segmentation architectures, namely: (i) *RefineNet* [28], (ii) *MobileNet-v2* [61], (iii) *DeepLab-v3* [30], (iv) *PSPNet* [29], (v) *HRNet-v2* [62], (vi) *SegFormer* [31], and (vii) *Mask2Former* [63]. To provide comparability between the different approaches, we trained all networks for 20 epochs with 5,000 iterations each and saved the weights after each epoch. Afterward, we selected the best weights by running the intermediate checkpoints on a validation set.

Our architecture achieves superior performance on both tested data sets (cf. Table III). We use the results of the *SkyFinder* data set and show the performance for each attribute class individually in Fig. 4. Compared to others, our approach performed significantly better under more challenging environmental conditions (e.g., night, fog). We believe that this is because our network explicitly learns the different sky appearances connected with environmental attributes.

TABLE III
RESULTS ON SKY SEGMENTATION PERFORMANCE

	SkyFinder		OUTSIDE15k	
Architecture	Acc. [%]	mIoU	Acc.[%]	mIoU
ResNet-50 [60]	79.21	0.771	71.25	0.691
MobileNetv2 [61]	78.04	0.786	70.49	0.711
RefineNet (RN-101) [28]	86.04	0.829	82.15	0.741
DeepLabv3 (RN-50) [30]	91.14	0.861	83.06	0.783
PSPNet (RN-50) [29]	91.54	0.878	81.15	0.732
HRNetV2-w48 [62]	92.04	0.891	84.05	0.781
SegFormer [31]	91.18	0.887	82.03	0.756
Mask2Former [63]	92.98	0.905	83.81	0.798
SkyCloudNet (RN-50)	**93.86**	**0.912**	**90.04**	**0.822**

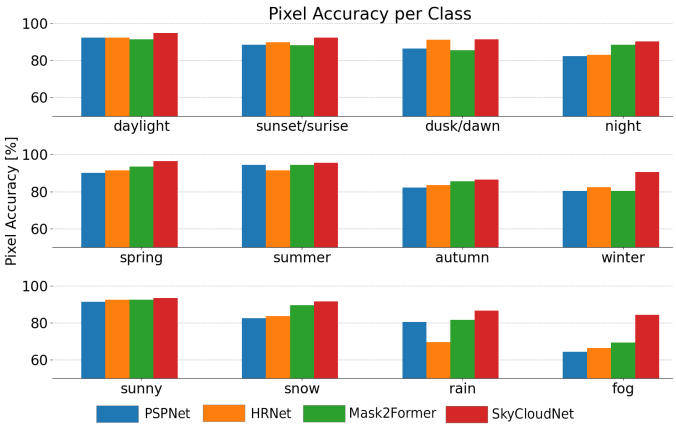

Fig. 4. Sky segmentation performance comparison per attribute class for selected approaches.

C. Cloud Segmentation

In this section, we evaluate the cloud segmentation performance of our network on the data set presented in Section III-B. We use the same evaluation metrics as for the sky segmentation (accuracy and *mean IoU*). For comparison, we use *HYTA* as a color feature-based approach and a *U-Net*-based approach [47] as deep learning alternative. In addition to the requirement of sky-only input, both approaches have additional limitations: *HYTA* is only capable of classification into sky and cloud pixels without further cloud subcategories, whereas the *U-Net* approach of Dev et al. [47] can distinguish between two types of clouds but requires fixed size input. Therefore, we performed two experiments with slightly different setups. For the first experiment, we masked the non-sky region for the *HYTA* method and changed the ground truth to only sky and cloud labels. For the second experiment, we kept the original labels but cropped a rectangular area from the sky region and used it as input for the method by Dev et al. [47]. To compare the results, we cropped the same regions from the prediction results that we acquired with *SkyCloudNet* on the whole images and calculated pixel accuracy and mIoU. The results of the two experiments are presented in Table IV and Table V. We show that our network performs better in both experiments. Fig. 6 depicts example images of cloud segmentation comparison with the *U-Net* approach from Dev et al. [47]. The first row

TABLE IV
CLOUD SEGMENTATION PERFORMANCE USING BINARY LABELS

Architecture	Acc. [%]	mIoU
HYTA [45]	61.12	0.393
SkyCloudNet	**84.56**	**0.671**

TABLE V
CLOUD SEGMENTATION PERFORMANCE USING CROPPED AREAS AS INPUT

Architecture	Acc. [%]	mIoU
Dev et al. [47]	69.23	0.451
SkyCloudNet	**82.96**	**0.637**

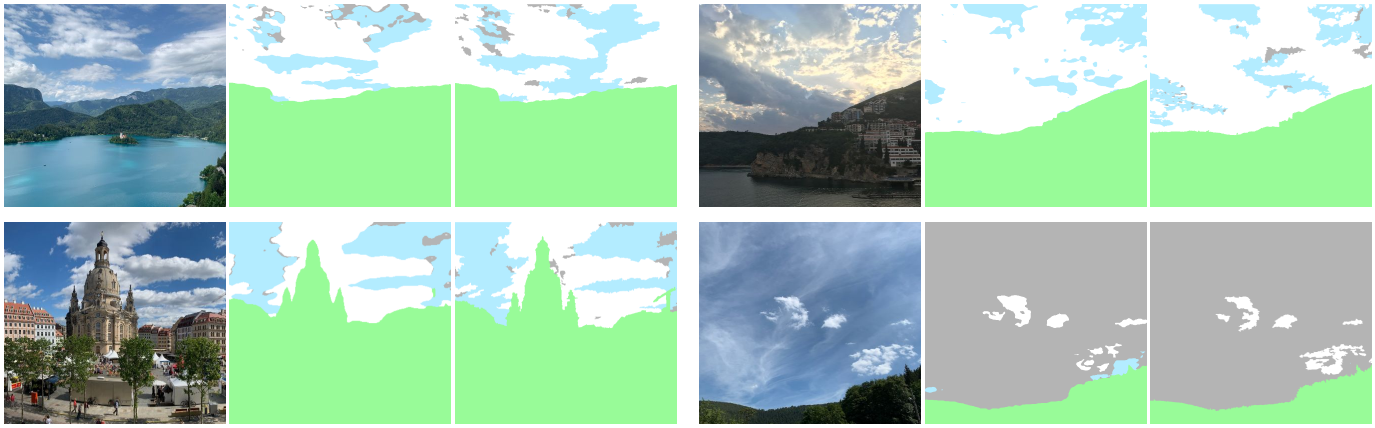

Fig. 5. Examples for sky & cloud segmentation using images from the *SkyCloud* data set. From left to right for each triple: input, prediction, ground truth. Labels: *green* - non-sky, *blue* - sky, *white* - thick cloud, *gray* - thin cloud.

shows that their approach works well when the sky and clouds match the typical color scheme (sky is blue, clouds are white). However, as shown in the second row, their approach fails under challenging lighting conditions, whereas *SkyCloudNet* manages to separate sky and cloud segments.

Finally, we evaluated the overall performance of our network. Due to the absence of alternative approaches, we only report the results of our architecture to foster and enable future research. Fig. 5 shows exemplary results for sky and cloud segmentation in different test scenes. Our network achieved an average pixel accuracy of 86.69% and a mean IoU of 0.73. With regard to the individual classes, the IoU results were: (i) *Ground* : 0.95, (ii) *Sky* : 0.73, (iii) *Thick cloud* : 0.68, and (iv) *Thin cloud* : 0.50. Thin clouds were the most difficult to detect as they often contain very small segments, and the distinction between thick clouds/sky is not trivial. Fig. 7 shows two typical error sources we encountered during the manual inspection. Fig. 7a shows an example where clouds were not detected due to the absence of light in the scene. This issue could possibly be addressed by increasing the number of classes during k-means clustering and adding specific cluster centroids for low-light scenes. However, the figure also shows that the sky segmentation works well even in such low-light situations. Fig. 7b indicates that the prediction can be insufficient when the cloud structure is very detailed. Approaches to mitigate this issue in the future could be to increase the input size of the network (increases memory consumption), divide the input image into patches, or process the image at multiple scales (both increase inference time).

Fig. 6. Detailed comparison of cloud segmentation results - from left to right: input, Dev et al. [47], *SkyCloudNet*, ground truth

VI. Ablation Study

We present the results of the test with different backbone network architectures in Table VI. The table also includes the segmentation results when training the network without feature forwarding. While we achieved the highest accuracy when using *ResNet-101*, the difference to *ResNet-50* was only marginal and therefore did not justify the increased memory requirements. With regard to the individual components of our network, the forwarding of environmental attributes accounts for an average accuracy improvement of 2.52%, while the region loss introduced in Section IV improved the accuracy by 1.5% in the sky segmentation task when using *ResNet-50* as the backbone.

Furthermore, as we use a composition of multiple data sets when training the neural network (cf. Section III), we wanted to investigate the isolated effect of each data set individually. Therefore, we trained the network with all possible combinations of the three data sets (*OUTSIDE15k* [50], *SkyFinder* [14] and *Transient Attributes*[19]. The results of this experiment are presented in Table VII and show that the combination of different data sets significantly improved

TABLE VI
Sky and Cloud Segmentation Performance Comparison on the *OUTSIDE15k* Data Set

	Sky Segm.		Sky & Cloud Segm.	
	Acc. [%]	mIoU	Acc. [%]	mIoU
MobileNetv2 [61]	80.23	0.713	72.45	0.625
ResNet-18	88.87	0.812	77.18	0.673
ResNet-50	90.04	0.822	86.69	**0.731**
ResNet-101	**90.12**	**0.829**	**86.75**	0.728
HRNetv2 [62]	89.88	0.838	84.21	0.692
ResNet-50 w/o feature forwarding	87.52	0.832	84.91	0.721
ResNet-50 w/o region loss	88.54	0.829	85.51	0.705
ResNet-101 w/o feature forwarding	87.66	0.834	84.95	0.728
ResNet-101 w/o region loss	88.81	0.829	85.55	0.709

TABLE VII
SKY SEGMENTATION PERFORMANCE COMPARISON USING THE *SkyCloud* DATA SET

	Acc. [%]	mIoU
Transient Attributes [19]	74.13	0.773
SkyFinder [14]	77.89	0.783
OUTSIDE15k [50]	81.23	0.789
SkyFinder + Transient Attributes	79.98	0.788
OUTSIDE15k + Transient Attributes	85.23	0.793
OUTSIDE15k + SkyFinder	88.85	0.806
OUTSIDE15k + SkyFinder + Transient Attributes	**90.56**	**0.845**

the overall performance, whereas the *OUTSIDE15k* data set has the greatest impact.

VII. LIMITATIONS AND TYPICAL ERRORS

Due to the absence of alternative data sets, we only evaluated attribute prediction of *SkyCloudNet* on the test set provided by Laffont et al. [19], which includes little variation in the scenes. To determine the applicability to more general images, a data set with human-generated attribute labels on a broader image data set is necessary.

Even though we aimed to create a cloud segmentation data set that covers various environmental conditions, a significant part of the *SkyCloud* data set was taken over a period of three months. Therefore, the variation in location, season, and weather conditions should be further improved in the future. However, since the cloud segmentation was trained in a semi-supervised way on a large number of images and different scenes, we assume that the results should transfer to other images.

A general limitation that applies to semantic segmentation is that ground-truth labels are inherently imperfect. Even after a two-round labeling process, they may contain ambiguous labels. Cloud labeling, as performed for our data set, is a highly challenging task because there is no discrete and distinct threshold between thin clouds and thick clouds. Thus, edge cases are often subjective.

VIII. CONCLUSION AND FUTURE WORK

With *SkyCloudNet*, we presented a semi-supervised multi-task neural network architecture for outdoor scene analysis: It addresses the tasks of high-level attribute estimation, sky segmentation, and cloud segmentation from natural outdoor images. Our architecture consists of a backbone and three separate heads that explicitly share information. We also present a data set for sky and cloud segmentation with 350 high-resolution images. *SkyCloudNet* outperforms alternative methods in the sky segmentation task (+0.88% accuracy). More noticeably, the cloud segmentation head trained on unlabeled data is far superior to existing approaches (+13.73% accuracy) in the cloud segmentation task. In the process, our overall framework eliminates the need for sky-only images in cloud segmentation.

In times of climate change and energy shortages, precise, fine-grained, and ubiquitous data are essential for scientific, regulatory, and executive entities to make informed decisions. *SkyCloudNet* promises to substantially contribute in this regard by enabling large-scale data collection on environmental attributes, sky, and clouds. Next to that, other applications like image stylization, or robot and drone navigation toolkits can benefit from the increased accuracy of the sky segmentation provided by *SkyCloudNet*.

In the future, we plan to investigate the applicability of *SkyCloudNet* in these use cases. In addition to that, we aim to add more scenes from currently underrepresented environmental conditions (e.g., *winter*, *rain*, or *fog*) to the *SkyCloud* data set. Also, we want to increase the number of classes in the cloud segmentation labels, at best matching cloud families or even types to further increase its applicability to real-life problems such as climate monitoring.

(a) Undetected clouds caused by low light in the scene.

(b) Details missing in fine cloud structures.

Fig. 7. Typical error cases in cloud segmentation. From left to right: input, prediction, ground truth

REFERENCES

[1] Y. Tsai, X. Shen, Z. Lin, K. Sunkavalli, and M. Yang, "Sky is not the limit: semantic-aware sky replacement," *ACM Trans. Graph.*, vol. 35, no. 4, 2016.

[2] K. A. Nice, J. S. Wijnands, A. Middel, J. Wang, Y. Qiu, N. Zhao, J. Thompson, G. D. Aschwanden, H. Zhao, and M. Stevenson, "Sky pixel detection in outdoor imagery using an adaptive algorithm and machine learning," *Urban Climate*, vol. 31, 2020.

[3] Z. Zou, R. Zhao, T. Shi, S. Qiu, and Z. Shi, "Castle in the sky: Dynamic sky replacement and harmonization in videos," *IEEE Trans. Image Process.*, vol. 31, 2022.

[4] G. L. Stephens, "Cloud Feedbacks in the Climate System: A Critical Review," *J. Clim.*, vol. 18, no. 2, pp. 237–273, jan 2005.

[5] W. Chu, K. Ho, and A. Borji, "Visual weather temperature prediction," in *WACV*. IEEE Computer Society, 2018.

[6] R. Müller, J. Trentmann, C. Träger-Chatterjee, R. Posselt, and R. Stöckli, "The role of the effective cloud albedo for climate monitoring and analysis," *Remote. Sens.*, vol. 3, no. 11, 2011.

[7] G. L. Stephens, J. Li, M. Wild, C. A. Clayson, N. Loeb, S. Kato, T. L'ecuyer, P. W. Stackhouse, M. Lebsock, and T. Andrews, "An update on earth's energy balance in light of the latest global observations," *Nature Geoscience*, vol. 5, no. 10, pp. 691–696, 2012.

[8] C. Kunjumon, S. S. Nair, D. Rajan S., P. Suresh, and S. Preetha, "Survey on weather forecasting using data mining," in *2018 Conference on Emerging Devices and Smart Systems (ICEDSS)*, 2018, pp. 262–264.

[9] M. Fathi, M. Haghi Kashani, S. M. Jameii, and E. Mahdipour, "Big data analytics in weather forecasting: A systematic review," *Archives of Computational Methods in Engineering*, pp. 1–29, 2021.

[10] S. Dev, Y. H. Lee, and S. Winkler, "Color-based segmentation of sky/cloud images from ground-based cameras," *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, vol. 10, no. 1, 2017.

[11] S. Dev, A. Nautiyal, Y. H. Lee, and S. Winkler, "Cloudsegnet: A deep network for nychthemeron cloud image segmentation," *IEEE Geosci. Remote. Sens. Lett.*, vol. 16, no. 12, 2019.

[12] G. Morales, A. Ramírez, and J. Telles, "End-to-end cloud segmentation in high-resolution multispectral satellite imagery using deep learning," in *2019 IEEE XXVI International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*. IEEE, 2019.

[13] S. Mohajerani and P. Saeedi, "Cloud-net: An end-to-end cloud detection algorithm for landsat 8 imagery," in *IGARSS*. IEEE, 2019.

[14] R. P. Mihail, S. Workman, Z. Bessinger, and N. Jacobs, "Sky segmentation in the wild: An empirical study," in *WACV*. IEEE Computer Society, 2016.

[15] O. Liba, Y. Movshovitz-Attias, L. Cai, Y. Pritch, Y. Tsai, H. Chen, E. Eban, and J. T. Barron, "Sky optimization: Semantically aware image processing of skies in low-light photography," in *CVPR Workshops*. Computer Vision Foundation / IEEE, 2020.

[16] N. Jacobs, W. Burgin, N. Fridrich, A. Abrams, K. Miskell, B. H. Braswell, A. D. Richardson, and R. Pless, "The global network of outdoor webcams: properties and applications," in *GIS*. ACM, 2009.

[17] N. Jacobs, N. Roman, and R. Pless, "Consistent temporal variations in many outdoor scenes," in *CVPR*. IEEE Computer Society, 2007.

[18] M. Islam, N. Jacobs, H. Wu, and R. Souvenir, "Images+ weather: Collection, validation, and refinement," in *IEEE Conference on Computer Vision and Pattern Recognition Workshop on Ground Truth*, vol. 6. Citeseer, 2013, p. 2.

[19] P. Laffont, Z. Ren, X. Tao, C. Qian, and J. Hays, "Transient attributes for high-level understanding and editing of outdoor scenes," *ACM Trans. Graph.*, vol. 33, no. 4, 2014.

[20] R. Baltenberger, M. Zhai, C. Greenwell, S. Workman, and N. Jacobs, "A fast method for estimating transient scene attributes," in *WACV*. IEEE Computer Society, 2016.

[21] C. Lu, D. Lin, J. Jia, and C. Tang, "Two-class weather classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, 2017.

[22] W. Chu, X. Zheng, and D. Ding, "Image2weather: A large-scale image dataset for weather property estimation," in *BigMM*. IEEE Computer Society, 2016.

[23] J. Tighe and S. Lazebnik, "Superparsing - scalable nonparametric image parsing with superpixels," *Int. J. Comput. Vis.*, vol. 101, no. 2, 2013.

[24] D. Hoiem, A. A. Efros, and M. Hebert, "Geometric context from a single image," in *ICCV*. IEEE Computer Society, 2005.

[25] Y. Liu, H. Li, and M. Wang, "Single image dehazing via large sky region segmentation and multiscale opening dark channel model," *IEEE Access*, vol. 5, 2017.

[26] G. De Croon, C. De Wagter, B. Remes, and R. Ruijsink, "Sky segmentation approach to obstacle avoidance," in *2011 Aerospace Conference*. IEEE, 2011, pp. 1–16.

[27] C. L. Place, A. U. Khan, and A. Borji, "Segmenting sky pixels in images: Analysis and comparison," in *WACV*. IEEE, 2019.

[28] G. Lin, A. Milan, C. Shen, and I. D. Reid, "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation," in *CVPR*. IEEE Computer Society, 2017.

[29] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *CVPR*. IEEE Computer Society, 2017.

[30] L. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *CoRR*, vol. abs/1706.05587, 2017.

[31] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," in *NeurIPS*, 2021.

[32] C. Sakaridis, D. Dai, S. Hecker, and L. V. Gool, "Model adaptation with synthetic and real data for semantic dense foggy scene understanding," in *ECCV (13)*, ser. Lecture Notes in Computer Science, vol. 11217. Springer, 2018.

[33] L. Sun, K. Wang, K. Yang, and K. Xiang, "See clearer at night: towards robust nighttime semantic segmentation through day-night image conversion," in *Artificial Intelligence and Machine Learning in Defense Applications*. SPIE, 2019.

[34] W. Zhou, A. Zyner, S. Worrall, and E. M. Nebot, "Adapting semantic segmentation models for changes in illumination and camera perspective," *IEEE Robotics Autom. Lett.*, vol. 4, no. 2, 2019.

[35] H. Porav, T. Bruls, and P. Newman, "I can see clearly now: Image restoration via de-raining," in *ICRA*. IEEE, 2019.

[36] H. Wang, Y. Chen, Y. Cai, L. Chen, Y. Li, M. Á. Sotelo, and Z. Li, "Sfnet-n: An improved sfnet algorithm for semantic segmentation of low-light autonomous driving road scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, 2022.

[37] A. Valada, J. Vertens, A. Dhall, and W. Burgard, "Adapnet: Adaptive semantic segmentation in adverse environmental conditions," in *ICRA*. IEEE, 2017.

[38] A. Pfeuffer and K. Dietmayer, "Robust semantic segmentation in adverse weather conditions by means of sensor data fusion," in *FUSION*. IEEE, 2019.

[39] A. Pfeuffer, K. Schulz, and K. Dietmayer, "Semantic segmentation of video sequences with convolutional lstms," in *IV*. IEEE, 2019.

[40] A. Pfeuffer and K. Dietmayer, "Robust semantic segmentation in adverse weather conditions by means of fast video-sequence segmentation," in *ITSC*. IEEE, 2020.

[41] A. Heinle, A. Macke, and A. Srivastav, "Automatic cloud classification of whole sky images," *Atmos. Meas. Tech.*, vol. 3, no. 3, pp. 557–567, 2010.

[42] J. Calbo and J. Sabburg, "Feature extraction from whole-sky ground-based images for cloud-type recognition," *Journal of Atmospheric and Oceanic Technology*, vol. 25, no. 1, pp. 3–14, 2008.

[43] M. Hasenbalg, P. Kuhn, S. Wilbert, B. Nouri, and A. Kazantzidis, "Benchmarking of six cloud segmentation algorithms for ground-based all-sky imagers," *Solar Energy*, vol. 201, pp. 596–614, 2020.

[44] S. Liu, L. Zhang, Z. Zhang, C. Wang, and B. Xiao, "Automatic cloud detection for all-sky images using superpixel segmentation," *IEEE Geosci. Remote. Sens. Lett.*, vol. 12, no. 2, 2015.

[45] Q. Li, W. Lu, and J. Yang, "A hybrid thresholding algorithm for cloud detection on ground-based color images," *Journal of atmospheric and oceanic technology*, vol. 28, no. 10, pp. 1286–1296, 2011.

[46] C. H. Li and C. K. Lee, "Minimum cross entropy thresholding," *Pattern Recognit.*, vol. 26, no. 4, 1993.

[47] S. Dev, S. Manandhar, Y. H. Lee, and S. Winkler, "Multi-label cloud segmentation using a deep network," in *2019 USNC-URSI Radio Science Meeting)*. IEEE, 2019, pp. 113–114.

[48] S. Dev, Y. H. Lee, and S. Winkler, "Multi-level semantic labeling of sky/cloud images," in *ICIP*. IEEE, 2015.

[49] S. Mohajerani, T. A. Krammer, and P. Saeedi, "A cloud detection algorithm for remote sensing images using fully convolutional neural networks," in *MMSP*. IEEE, 2018.

[50] C. Gerhardt, F. Weidner, and W. Broll, "OUTSIDE: multi-scale semantic segmentation of universal outdoor scenes," in *MMSP*. IEEE, 2021.

[51] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *CVPR*. IEEE Computer Society, 2017.

[52] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, 2015.

[53] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, and A. Torralba, "Object detectors emerge in deep scene cnns," in *ICLR*, 2015.

[54] X. Li, T. Lai, S. Wang, Q. Chen, C. Yang, R. Chen, J. Lin, and F. Zheng, "Weighted feature pyramid networks for object detection," in *ISPA/BDCloud/SocialCom/SustainCom*. IEEE, 2019.

[55] Y. Ouali, C. Hudelot, and M. Tami, "Autoregressive unsupervised image segmentation," in *ECCV (7)*, ser. Lecture Notes in Computer Science, vol. 12352. Springer, 2020.

[56] J. H. Cho, U. Mall, K. Bala, and B. Hariharan, "Picie: Unsupervised semantic segmentation using invariance and equivariance in clustering," in *CVPR*. Computer Vision Foundation / IEEE, 2021.

[57] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with gpus," *IEEE Trans. Big Data*, vol. 7, no. 3, 2021.

[58] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*. IEEE Computer Society, 2009.

[59] J. C. Spall, *Introduction to Stochastic Search and Optimization*. Hoboken, NJ, USA: John Wiley & Sons, Inc., mar 2003.

[60] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*. IEEE Computer Society, 2016.

[61] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *CVPR*. Computer Vision Foundation / IEEE Computer Society, 2018.

[62] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, 2021.

[63] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *CVPR*. IEEE, 2022.