LumNet: Learning to Estimate Vertical Visual Field Luminance for Adaptive Lighting Control

PRINCE U.C SONGWA, Eindhoven University of Technology, Netherlands

AAQIB SAEED, Eindhoven University of Technology, Netherlands

SACHIN BHARDWAJ, Eindhoven University of Technology, Netherlands

THIJS W. KRUISSELBRINK, Eindhoven University of Technology, Netherlands

TANIR OZCELEBI, Eindhoven University of Technology, Netherlands

High-quality lighting positively influences visual performance in humans. The experienced visual performance can be measured using desktop luminance and hence several lighting control systems have been developed for its quantification. However, the measurement devices that are used to monitor the desktop luminance in existing lighting control systems are obtrusive to the users. As an alternative, ceiling-based luminance projection sensors are being used recently as these are unobtrusive and can capture the direct task area of a user. The positioning of these devices on the ceiling requires to estimate the desktop luminance in the user's vertical visual field, solely using ceiling-based measurements, to better predict the experienced visual performance of the user. For this purpose, we present LUMNET, an approach for estimating desktop luminance with deep models through utilizing supervised and self-supervised learning. Our model learns visual representations from ceiling-based images, which are collected in indoor spaces within the physical vicinity of the user to predict average desktop luminance as experienced in a real-life setting. We also propose a self-supervised contrastive method for pre-training LUMNET with unlabeled data and we demonstrate that the learned features are transferable onto a small labeled dataset which minimizes the requirement of costly data annotations. Likewise, we perform experiments on domain-specific datasets and show that our approach significantly improves over the baseline results from prior methods in estimating luminance, particularly in the low-data regime. LUMNET is an important step towards learning-based technique for luminance estimation and can be used for adaptive lighting control directly on-device thanks to its minimal computational footprint with an added benefit of preserving user's privacy.

CCS Concepts: • **Computing methodologies** → **Supervised learning by regression**; **Neural networks**; **Unsupervised learning**; • **Human-centered computing** → *Ambient intelligence*.

Additional Key Words and Phrases: luminance estimation, adaptive lighting, deep learning, self-supervised learning, HDR, ambient intelligence

ACM Reference Format:

Prince U.C Songwa, Aaqib Saeed, Sachin Bhardwaj, Thijs W. Kruisselbrink, and Tanir Ozcelebi. 2021. LumNet: Learning to Estimate Vertical Visual Field Luminance for Adaptive Lighting Control. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 2, Article 79 (June 2021), 20 pages. <https://doi.org/10.1145/3463500>

Authors' addresses: Prince U.C Songwa, Eindhoven University of Technology, Eindhoven, Netherlands; Aaqib Saeed, Eindhoven University of Technology, Eindhoven, Netherlands; Sachin Bhardwaj, Eindhoven University of Technology, Eindhoven, Netherlands; Thijs W. Kruisselbrink, Eindhoven University of Technology, Eindhoven, Netherlands; Tanir Ozcelebi, Eindhoven University of Technology, Eindhoven, Netherlands, t.ozcelebi@tue.nl.

This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2021 Copyright held by the owner/author(s).

2474-9567/2021/6-ART79

<https://doi.org/10.1145/3463500>

1 INTRODUCTION

The effects of light on human health and well-being have been the subject of substantial research for several years. It has been shown that light influences the human circadian and circannual rhythms [55], alertness [29], mood and vitality [41] and can be used in the treatment of sleep disorders, such as jet lag [2]. In a work environment, a light has been shown to improve the productivity of workers and reduce failures, accidents and absenteeism [55]. Moreover, lighting also positively influences human visual performance. High quality lighting improves a person's visual performance and enables them to better accomplish their tasks [58]. Therefore, it is essential to be able to measure and improve visual performance. It has been measured in previous research using illuminance distributions [3, 4, 35, 45]. Illuminance refers to the measure of the amount of light that falls on a surface, as shown in Figure 2. It is measured in lux (lx). Along these lines, it has been used to measure visual performance because it is inexpensive and easy to measure. However, it has been shown to demonstrate high individual variability, with no consensus being found among users, about the experienced illuminance levels [57]. Recent research has shown that visual performance can be better predicted using luminance distributions [56]. The luminance refers to the measure of the amount of light that is reflected or emitted from the projected surface. It is measured in candelas/square meter (cd/m^2). The desktop luminance is the amount of light reflected or emitted from the desktop area in an indoor space, relative to the measuring point. The luminance-based metrics can better predict the visual performance and human subjective lighting preferences than measures that rely on illuminance because luminance is more closely related to brightness as perceived by the humans [9]. Traditionally, luminance was difficult to measure but this has been made easier with HDR imaging [25].

The luminance distribution can be estimated from the Red-Green-Blue (RGB) pixel values of the High Dynamic Range (HDR) images [18]. Generally, it is measured at the eye level using luminance cameras and is, therefore, a suitable input for adaptive lighting control systems [26]. However, various practical issues ensue when such systems are deployed in real-world environments. These systems use classical strategies for directly measuring the field of view of the user, such as mounting the luminance camera on a PC monitor in an office or placing it in the vicinity of the user with minor angular and translational variations with respect to the vertical visual field of the user [36]. The visual field refers to the portion of space that one can see at the same time without moving one's eyes or head. The vertical visual field (VVF) refers to the visual field in the vertical direction. This corresponds to 135° , specifically 60° upwards and 70° downwards [52]. Systems which use these classical strategies have the main drawback that the cameras can be in the way, interfering with daily user activities and might also omit the user's direct task area [28]. The most suitable camera position has been shown to be on the ceiling, where the cameras are not obtrusive and can capture the direct task areas of all users. The convenience of positioning the luminance cameras on the ceiling has the disadvantage that it creates the need to translate the ceiling-based luminance measurements to the VVF of the users in order to estimate the luminance as perceived by these users. This is because the ceiling-based cameras are not in the VVF of the users. The feasibility of estimating the vertical visual field desktop luminance (VVF DL) from ceiling-based measurements was examined in [27]. The prior work showed that ceiling-based measurements can reasonably estimate vertical visual field luminance measurements based on linear models which approximate the relation between the measurements from these positions, with normalized root mean square errors ranging from 10.3% to 14.3% with an average bias of 14.2%. The approach presented here uses a learning-based strategy to estimate the VVF DL from ceiling-based measurements (i.e., raw images) with deep neural networks.

Over the last decade, deep learning has revolutionized several domains to match human-level performance from image recognition to competitive game-playing. Particularly, convolutional neural networks (CNNs) [30] have been widely adopted for visual recognition tasks, such as object detection [46], semantic segmentation [31], and captioning [59] for learning high-level representation from data in an end-to-end manner. However, majority of the success is attributed to supervised learning that requires well-annotated data and it is costly and tedious

Fig. 1. Overview of the luminance estimation system for adaptive lighting control. The Bee-Eye cameras placed at the eye level of the users can be used to measure the average desktop luminance as experienced in a real-world setting. However, the ceiling-based cameras are preferable in practice as they are not obtrusive and they can cover the direct task areas of users. We propose to use a learnable module that can utilize ceiling-based cameras to measure and translate to the vertical visual field for estimating the average desktop luminance with a deep convolutional neural network, named LumNet. The ceiling-based images are HDR images, which are generated by merging low dynamic range images of the users' task areas, from multiple exposures. The VVF₂ and VVF₃ positions refer to positions in the users' vertical visual fields, as described in Section 4.1.

to obtain in real-world scenarios. Therefore, deep unsupervised learning (or learning from unlabeled input) is recently becoming the subject of interest, especially self-supervised learning (SSL). In this realm, the network is trained to solve a pretext task in order to learn high-level representations without supervision that can be used to solve end-tasks of interest with minimal labeled data. In the last few years, SSL has been extensively studied for learning from a wide-variety of modalities, such as for images [1, 6, 39, 65] and videos [49, 60], text [11, 39], and others [47]. Motivated by the success of SSL, we pose the question, whether VVFDL can be estimated efficiently with a deep model learned directly from data, potentially in a self-supervised manner to reduce dependence on human provided labels.

In this paper, we propose LUMNET, a deep convolutional neural network based luminance estimation technique that uses ceiling-based measurements and it can be incorporated in adaptive lighting control systems. We examine the performance of LUMNET in predicting the VVFDL when trained on ceiling-based images, which are collected in the physical vicinity of the user. We also examine LumNet's ability to generalize when fine-tuned on

Fig. 2. Illuminance vs Luminance. The arrows incident on the surface represent the Illuminance (lux), which is a measure of the amount of light incident on the surface. The arrows leaving the surface represent the luminance (candelas/m²), a measure of the amount of light that is reflected or emitted from the projected surface.

out-of-domain (unseen) data using transfer learning. Finally, we propose a self-supervised approach based on contrastive learning to pre-train the model with unlabelled data. In our SSL approach, we use a contrastive loss similar to [6, 17, 39] but without relying on input transformations for solving instance discrimination pretext task to learn high-level representations from unlabeled data for the end-task of luminance estimation. The learned self-supervised model is fine-tuned on a small labeled dataset to minimize the requirement of costly semantic annotations. We demonstrate that LUMNET achieves a better performance than prior methods based on domain-knowledge [27, 28] when trained on labeled data consisting of ceiling-based images. This also holds when LUMNET representations learned from a labeled data are transferred for fine-tuning on a smaller dataset consisting of different VVF positions. Furthermore, LUMNET trained with our self-supervised technique also achieves a better overall performance than the baseline results achieved in [27, 28]. At the moment of conducting our research, no prior work on luminance estimation was based on neural networks. LUMNET is therefore a novelty when it comes to luminance estimation.

In summary, the key contributions of our work are as follows:

- We propose LUMNET, a novel luminance estimation technique based on deep convolutional neural networks that uses ceiling-based visuals to measure luminance as a user experience at the eye-level.
- We introduce a self-supervised approach based on contrastive learning, which is able to achieve a better performance with low-labeled data, than the baseline results.
- We examine LUMNET's performance in predicting the VVFDL when trained on ceiling-based images, which are collected in the physical vicinity of the user in real-life setting.
- We evaluate LUMNET's ability to generalize on out-of-domain (unseen or collected in a different environment) data using transfer learning.
- Furthermore, the LUMNET can be incorporated in adaptive lighting control systems for on-device inference.

The rest of the paper is structured as follows. The related work in the fields of luminance distribution measurements and visual representation learning are discussed in Section 2. Section 3 presents an overview of the proposed approach, including a description of LumNet's architecture, transfer learning and instance discrimination based self-supervised contrastive objective. The data pre-processing, experiments, and the results are presented in Section 4. We provide a discussion of the findings in Section 5 before concluding in Section 6.

2 RELATED WORK

In this section, we discuss relevant work in the field of luminance measurements and learning visual representations with deep neural networks. We also discuss studies on transfer learning and unsupervised (self-supervised) learning as our key contributions involve these concepts as well.

2.1 Quantifying Luminance

Luminance metrics can predict human subjective lighting preferences and most importantly, visual performance. Therefore, it is important to estimate it in the VVF of users. Doulos et al. [12] and Motamed et al. [37] showed that luminance measurements can be performed from a ceiling position using charge-coupled device (CCD) and luminance cameras. However, the ceiling-based measurements are not in the vertical visual field (VVF) of the user and therefore they do not reflect luminance as experienced by the user. Kruisselbrink et al. [27] conducted a study to examine the feasibility of estimating the VVFDL from ceiling-based measurements. They measured the average desktop luminance under several conditions using luminance cameras, which were positioned both on the ceiling and in the VVF of the user. An approximate relation between the measurements from these two positions was extracted using a linear model. However, their results showed that the VVFDL cannot be exactly replicated through using ceiling-based measurements but a reasonable estimate can be provided. Kruisselbrink et al. [28] also conducted a study to determine the most suitable ceiling-based position for measuring luminance distributions for luminance-based metrics including desktop luminance. Their study showed that the most suitable ceiling-based position was above the aisle at an angle of 20° relative to the ceiling. They also evaluated the estimation of the vertical visual field luminance from measurements with the same procedure and showed that good estimations could be achieved. Similarly, a linear model based on domain-knowledge was used to establish the relation between the estimated VVFDL and the actual VVFDL. The resulting linear model together with the mathematical model in [27] (also a linear model) were used to calculate VVFDL values. These results serve as a baseline for the proposed learning based luminance estimation i.e., LUMNET, a deep convolutional network.

2.2 Visual Representation Learning

Convolutional neural networks have been used to learn visual representations from images using both supervised and unsupervised learning settings. In supervised representation learning realm, the success of CNNs dates back to Krizhevsky et al. [24], as their model was used for image classification on the ImageNet ILSVRC-2010 dataset [10]. Since then, CNNs have been widely adopted for solving challenging problems in computer vision ranging from object detection [46], action recognition in videos [14], learning to play Atari from raw pixel values [34], and medical imaging. Rajpurkar et al. [44] developed CheXNet, a convolutional model which is used to detect pneumonia from chest X-rays. CheXNet was trained with images from the ChestX-ray14 dataset [61], with labels of thoracic diseases, including pneumonia. Likewise, Gulshan et al. [16] developed a CNN to detect diabetic retinopathy and diabetic macular edema in retinal fundus images from retinal images. Moreover, supervised learning has also been used for object recognition using variants of their CNN architecture with various depths, trained on the ImageNet dataset [46, 51]. Sermanet et al. [48] developed a CNN-based framework, called *Overfeat*, for computer vision tasks such as classification, localization and detection. It has also been used in Razavian et al. [50] to extract generic features and to apply them to diverse tasks such as scene recognition, attribute detection, fine grained recognition and image retrieval on multiple datasets. In all of these approaches, the models were trained with annotated data, which is scarce in real-world scenario. Transfer learning has been used as a solution to tackle the scarcity of annotated data. It seeks to achieve the utilization of learned knowledge with deep neural networks from one domain (a source) to another domain (a target) for improving the generalization, when annotated data is limited [54]. It has been widely adopted for learning visual representations from pre-trained deep models both in supervised [32, 50] and unsupervised learning regimes [5, 21, 22, 43]. Over the recent years,

unsupervised learning gains increasing attention to exploit the availability of large-scale unlabeled data to learn generalizable features. Particularly, self-supervised learning is a promising subarea of unsupervised learning that is achieving state-of-the-art performance on vision benchmarks. It formulates a pretext task for the network to solve where labels can be acquired from the data itself [6, 15, 17, 23, 39, 49, 64]. Here, the transfer learning approach of pre-training a model on a large-scale unlabeled dataset with an auxiliary objective and transferring on another has also been investigated in [1, 6, 17]. We follow a similar evaluation protocol for assessing the quality of learned features through either using a linear regression or fine-tuning the entire network. Importantly, we utilize domain-specific datasets for learning luminance estimation models as opposed to general image recognition corpus.

3 METHODOLOGY

We develop a framework for luminance estimation for adaptive lighting control with a deep neural network, LUMNET, which can be trained with both labeled and unlabeled instances consisting of ceiling-based images. In this section, we start with describing our approach for estimating the VVFDL, as well as the data augmentation operations suitable for LUMNET. Next, we provide details of network's architecture and how it is used to learn and transfer high-level visual representations between domain-specific datasets. Lastly, we describe our self-supervised contrastive learning approach that can be used to learn useful representations entirely in an unsupervised manner without requiring human supervision.

3.1 Learning to Estimate Luminance

In order to measure the visual performance of users in an indoor environment (e.g., an office space), the luminance distributions in their respective vertical visual fields can be generally measured using Bee-Eye cameras [26], which are positioned at the users' eye level. This position, however, causes interference of the cameras with the users' daily activities. In order to avoid this issue, the cameras are placed on the ceiling, where they are not obtrusive. An overview of such a setup for measuring luminance distributions is illustrated in Figure 1. The luminance distributions can be computed directly from the HDR images of the ceiling-based cameras. However, the computed values directly do not provide a good estimate of the luminance distribution as experienced by the user, since, these measurements are not directly taken in the VVF of the user. In addition, they are situated in a rather complex environment with changing daylight conditions. Therefore, a translation of the ceiling-based measurements to the VVF of the users is required, in order to accurately estimate the luminance distribution to match a user's experience. Towards this objective, Kruisselbrink et al [27, 28] examined the feasibility of estimating the VVFDL using linear modeling. Our goal is to determine whether deep learning, in particular deep convolutional networks can be used to estimate the VVFDL from ceiling-based measurements to improve over the domain-knowledge driven (or hand-crafted technique) baselines [27, 28] through harnessing data-driven (or learned models) strategies. Our approach consists of three key aspects.

- (i) Firstly, learning a luminance estimation model, LUMNET with labeled instances from datasets comprising ceiling-based images which are collected in the physical vicinity of the user. These are the Kruisselbrink Luminance Estimation Datasets (KLED), which are further referred to as KLED-2020 [28] and KLED-2019 [27] datasets. Further details about these datasets can be found in Section 4.1.
- (ii) Secondly, evaluating LUMNET's ability to generalize well on an unseen (out-of-domain or different environment) data through fine-tuning its pre-trained weights using transfer learning.
- (iii) Lastly, effectively using unlabeled data for self-supervised contrastive learning of LUMNET and fine-tuning it on above mentioned datasets for the downstream task of luminance estimation. It is to evaluate the model generalization when trained with unannotated data.

Fig. 3. An HDR image along with the corresponding luminance mask. An original HDR image captured at the user’s eye-level is pre-processed (cropped) to remove the borders. The luminance mask is extracted from it with a focus on desktop area for which the mask has a value of one, while all other pixels are set to zero. The mask is used to compute luminance for the proposed deep learning based estimation system.

For LUMNET input we compute the desktop luminance for the HDR images at each VVF position as $L = \alpha \times R + \beta \times G + \gamma \times B$, where $\alpha = 0.2125$, $\beta = 0.7125$, and $\gamma = 0.0721$ and take the average of all pixels. α , β and γ represent coefficients for transforming pixel values from the RGB color space to the Y primary of the CIE XYZ color space, since the Y primary is used to indicate luminance [18, 19]. The pixels which are used for this computation are those in the desktop area of the HDR image. We use luminance masks to set the pixel values of all others to zero, such that only the pixels in the desktop area are used to determine the luminance as depicted in Figure 3. It is done by multiplying the each pixel’s value in the luminance mask with the value of the corresponding pixel in the VVF HDR image.

We convert the ceiling-based HDR images into the portable network graphics (PNG) format before feeding them to LUMNET as it is widely supported by existing tools. These images contain dark borders (see Figure 3) and therefore cropping is performed to ensure that the focus of the model is solely on the image itself. An additional benefit is that, it greatly lowers the use of memory and computation resources as the number of pixels needed for the convolution operations is reduced. Furthermore, we apply min-max normalisation to the training and test samples of the computed average desktop luminance values to scale the output within the range $[0, 1]$. This normalisation is suitable for the sigmoid activation function, which has a range of $(0, 1)$ and it is used in the output layer of the model (section 3.3 for architecture details). LUMNET is trained end-to-end with either labeled inputs (or unannotated for self-supervised pre-training) consisting of the cropped ceiling-based HDR images, which has been converted to PNG, and average desktop luminance values of their corresponding VVF positions. LUMNET learns representations from the input images and predicts the luminance for each VVF position. These predictions are used to compute the root mean squared error (RMSE) between the predicted and the actual average desktop luminance values. The RMSE is used as an evaluation metric to assess the performance of proposed method.

3.2 Data Augmentation

Given that the KLED-2020 considered datasets are limited in size, four data augmentation operations are used to generate synthetic data on-the-fly during learning phase to improve robustness of LUMNET. The augmentations include the brightness, Gaussian blur, vertical and horizontal shifts operations. Data augmentation generates additional data-points to cover a wide-variety of input transformations when the dataset is of limited size to reduce the risk of overfitting. These operations are applied randomly on the dataset during training, to produce a mix of augmented and original image batches. The brightness augmentation operation consists of adding brightness noise to some images randomly. It is suitable for luminance measurements because the brightness

conditions of the environment changes throughout the day based on natural (and artificial) lighting. We also randomly apply the Gaussian blur with a specific blur radius. This operation is also suitable as some images might be blurred in practice, due to aberrations of the Bee-Eye camera lenses which can result in out-of-focus images. The presence of dust or other particles on the camera lenses could also affect the quality of the resulting image. Similarly, we apply the vertical and horizontal shifts, where the image is shifted by a percentage of its original height and width, respectively. This is suitable because in practical scenarios the luminance cameras might not always be fixed in the exact same position.

3.3 LumNet Architecture Design

Our model takes a pre-processed ceiling-based image of size 128×128 pixels as input and outputs luminance value predictions for the three user positions as described in Section 4.1. LUMNET consists of 2 blocks each with 2 convolutional layers of a kernel size 3×3 followed by maxpooling layer with pooling size of 2×2 and a stride of 1 for downsampling. We flatten the output of the last convolutional layer before feeding into a fully-connected layer with 512 hidden units. We use rectified linear unit as activation in all layers except the last, which has a sigmoid activation to produce a normalized output. The output layer has 3 hidden units to produce luminance values corresponding to the VVF₁, VVF₂ and VVF₅ positions, respectively. Importantly, as KLED-2019 dataset has data collected from positions VVF₃ and VVF₂ only, we adapt the last layer of the model to have 2 hidden units to produce estimate for the corresponding locations. Note that in our case, the network has 3 outputs to closely estimate the luminance from various viewpoints. In a real-life setting the average of these could be used to adapt lighting. Likewise, it can be extended in a straightforward manner through increasing hidden units in the last layer to quantify the luminance from other angles if required. The high-level overview of the LUMNET architecture is illustrated in Figure 4 and we highlight that due to relatively small size of our model, it can be directly deployed on resource-constrained devices (such as Raspberry-Pi) for efficient inference while preserving user's privacy.

3.4 Transferring Learned Representations

It is been shown that the initial layers of a CNN learn generic features common to images, whereas, the final layers extract task-specific features [50]. To achieve knowledge transfer, we adopt two strategies. In the first scheme, we train the pre-trained model on the target dataset without freezing any layers and use a smaller learning rate for fine-tuning. Specifically, we pre-train LUMNET on the KLED-2019 dataset, then train and evaluate it on the KLED-2020 dataset using the specifications mentioned in Section 4.2 and with a small learning rate of $5e^{-4}$. In the second scheme, we freeze all the layers of a pre-trained model except the last layer, then we train the model on the target dataset. Specifically, we pre-train LUMNET on unlabeled data from the KLED-2019 dataset, freeze all the layers except the last, then train and evaluate it on labeled data from the KLED-2020 dataset, using the settings mentioned in Section 4.2 and a learning rate of $5e^{-3}$.

3.5 Self-Supervised Contrastive Learning

Self-supervised learning aims to learn general-purpose representation from unlabeled data through solving a auxiliary task for which the labels can be acquired from the data itself. The goal of solving the auxiliary task is to learn image representations, which can be useful in solving the downstream tasks of interest. The auxiliary task is also referred to as a pretext or surrogate task. Here, we use contrastive learning objective similar to MoCo [17] and SIMCLR [6] to learn high-level representation from unannotated data as acquiring large amount of labeled (with luminance) instances is expensive and time-consuming in real-world setting. For contrastive learning, we use an instance discrimination [53, 63] approach, which seeks to learn image representations that capture similarities between related instances. Specifically, a given example, referred to as the *query*, is associated to

Fig. 4. LumNet Architecture.(a) The ceiling-based HDR images are converted to PNG and cropped to remove the dark borders, before being fed into the model. The model output consists of estimated luminance values for each of the VVF positions depending on the dataset/task. (b) The deep convolutional neural network for learning to estimate luminance directly from ceiling-based images. Our model has low computational requirement for inference and it can be used on a resource constrained devices (e.g., Raspberry Pi) directly along with the HDR camera for estimating luminance.

related samples, referred to as *positives* and separated from dissimilar samples referred to as *negatives* in the feature space. We use the InfoNCE loss [39] to learn embeddings that capture such high-level similarities. This loss function maximizes the mutual information between examples (see [39] for theoretical proof). The similarity between these samples is measured using the cosine similarity as recent work [38], which is the dot product of their normalised vector representations. These vectors are K-dimensional and are defined for the query, positive and negative samples as $\mathbf{v}$, $\mathbf{v}^+$, and $\mathbf{v}^- \in \mathbb{R}^K$, respectively. Similar to Park et al. [40], these vectors are normalized to prevent the latent space from expanding or collapsing. The computed dot products are scaled by a temperature parameter τ which is set to 0.2 and the resulting scaled distances are used as logits which are fed to standard cross-entropy loss for optimizing network parameters, which is defined as follows for a batch of I samples:

$$\mathcal{L}(\mathbf{v}, \mathbf{v}^+, \mathbf{v}^-) = -\log \left[\frac{\exp(\mathbf{v} \cdot \mathbf{v}^+ / \tau)}{\exp(\mathbf{v} \cdot \mathbf{v}^+ / \tau) + \sum_i^I \exp(\mathbf{v} \cdot \mathbf{v}_i^- / \tau)} \right] \quad (1)$$

Fig. 5. Illustration of pseudo-classes for contrastive learning from unlabeled data. To generate anchor and positive samples for self-supervised learning, we create pseudo classes as opposed to utilization of augmentation as done generally for standard image recognition tasks. Here, each class represents the sky conditions of the particular hour of the day on which the picture was taken, as defined by the KNMI.

Our pretext task can also be considered as a multi-class classification problem, where, the unlabeled data is annotated with pseudo-labels representing 3 classes. The similarity between the instances of the specified classes is measured using LUMNET in the embedding space. The classes are defined as *clear*, *intermediate* and *overcast*. They represent the sky conditions of the hour of the day on which the images were taken (in a mock-up office location), as determined by the Royal Netherlands Meteorological Institute (KNMI). The KNMI provides a cloud cover indicator (N), which is an integer between 1 and 9. This indicator is used to define the classes as: $N < 3 \Rightarrow \text{clear}$, $3 \leq N < 8 \Rightarrow \text{intermediate}$, $N = 8 \Rightarrow \text{overcast}$, and $N = 9 \Rightarrow \text{invisible sky condition}$. The invisible sky condition is not applicable in our setting because no image was taken under this cloud condition. The metadata of our utilized datasets (see section 4.1) contains timestamps for each of the image. These timestamps were attributed automatically by the Bee-Eye cameras when generating the images, therefore, it is possible to determine the day and time of the image capture. We use this information to map each image to one of the three pseudo-classes defined earlier, based on the hour of the day on which the image was taken. We opt for this strategy to generate examples (i.e., query and positive pairs) as opposed to utilizing data augmentation as in some recent self-supervised work [6, 17] since heavy input transformation could distort the lighting or other general-purpose features. Figure 5 provides a random sample of pre-processed images from the KLED-2019 dataset, each mapped to its respective class. Furthermore, for contrastive learning, we slightly alter the architecture of LUMNET by adding a global average pooling layer after the convolution blocks, to aggregate the features and reduce the number of parameters to learn. We train LUMNET model with the InfoNCE [39] loss with a batch size of 1000 with a learning rate of $5e^{-4}$ for a maximum of 20 epochs. We sample mini-batches of (anchor, positive) pairs but do not perform explicit negative sampling as other examples in the batch act as negatives for a particular instance. Each pair (anchor and positive) is randomly chosen from one of the three classes mentioned above (clear, intermediate, overcast) and constitute the input to the network for optimizing the InfoNCE loss [39]. For initial assessment of the learned representations with a contrastive loss, we use an image retrieval task. Figure 6 shows the 10 nearest neighbor examples for each query image in decreasing order of similarity. It shows that the network learns high-level features relevant to distinguish instances based on different lighting conditions.

4 EXPERIMENTS

We demonstrate the performance of LUMNET when trained with labeled data on KLED-2020 and KLED-2019 datasets. We demonstrate that our learning based approach significantly improves upon the existing baselines in

Fig. 6. Assessment of contrastive learning with an image retrieval task. We evaluate the network’s capability in learning high-level features with self-supervised objective in a qualitative manner through a task of retrieving similar images from the dataset. For each randomly selected query image (on the left) we provide 10 nearest neighbor examples in decreasing order of similarity (in the embedding space) as provided by the LUMNET model.

VVFDL estimations. We also show that LUMNET generalizes well on unseen data, when fine-tuned end-to-end with transfer learning. Finally, we examine the performance of LUMNET when pre-trained with unlabelled data with contrastive learning and using learned representations on downstream task through fine-tuning on labeled data.

4.1 Dataset and Pre-Processing

For learning to estimate user-experienced luminance from ceiling-based cameras. We use 2 datasets, namely, KLED-2020 and KLED-2019. These datasets contain coupled HDR images, in a resolution of 901 x 676 pixels, measured from a ceiling-based position and from multiple VVFs of the users. The important characteristics of the datasets are illustrated in Table 1. All measurements are conducted with identical luminance cameras referred to as the Bee-Eye [26]. The Bee-Eyes utilize a Raspberry Pi 3 Model B and a Raspberry Pi Camera Board version 2 and a miniature fish eye lens. Each HDR image is built, using hdrngen [62], on seven individual exposures ranging from 9 μ s to 2s. The ceiling-based cameras are placed at position C1 on the ceiling, as shown in Figure 7. Both datasets are collected in the same office environment as illustrated in Figure 7. The KLED-2020 contains three VVFs at a height of 1.2m in front of the desktop area. Additionally, these desktop areas are monitored by a ceiling-based Bee-Eye above the aisle with an orientation of 20° relative to the ceiling. The window opening with a height of 1.35m covers the entire west facade. The second dataset, KLED-2019, has 2 VVFs with a ceiling-based Bee-Eye directly above the desktops. The west facade was covered completely by a window with a height of 1.9m. The images in each dataset were captured at 10 minutes interval from 5:30 to 22:00 for the KLED-2020 dataset, and from 8:30 to 12:00, and 13:00 to 16:30 for the KLED-2019 dataset. Based on the R, G and B values of the HDR images the luminance (L) is calculated for each individual pixel as $L = \alpha \times R + \beta \times G + \gamma \times B$, where $\alpha = 0.2125$, $\beta = 0.7125$, and $\gamma = 0.0721$. Subsequently, the desktop luminance was extracted for each individual Bee-Eye relative to the VVFs by gathering and averaging the luminance value of the respective pixels. We mask out parts of the image such that only the desktop area for the desired position is visible, following [28].

Table 1. Key characteristics of the datasets.

	KLED-2020 [28]	KLED-2019 [27]
Collection Time	5:30 to 22:00	8:30 to 12:00, 13:00 to 16:30
Interval	10 minutes	10 minutes
Resolution	901 × 676	901 × 676
Ceiling-based positions	C1	C1
Orientation ceiling-based	20°	90°
Vertical visual field positions	VVF ₁ , VVF ₂ , VVF ₅	VVF ₂ , VVF ₃
Total HDR images	1604	4416

4.2 Network Training and Optimization Details

We train LUMNET for 100 epochs using the Adam optimizer[20] with a learning rate of $5e^{-4}$, and the binary cross-entropy loss function. This loss function has been shown in [7] to be suitable for training autoencoders on image data, where the pixel values are normalised within the range [0, 1]. It is therefore a suitable option for training LUMNET, given that the sigmoid activation function in the output layer of LUMNET restricts the output within a range of [0, 1]. The Min-Max normalisation [42] is also applied, to scale the luminance values within the range [0, 1] to be compatible with network output. The use of the binary cross-entropy loss also resulted in a better performance than other loss functions, namely, mean squared error (MSE), mean absolute error (MAE), and Huber loss [33] that we explored earlier. In order to estimate the model performance, we use 10-folds cross-validation. The dataset is randomly split into a training set (80%) and a validation set (20%), for each fold. It reduces the risk of overfitting and selection bias, given that LUMNET is trained and tested using different subsets of each dataset. For the KLED-2020 dataset, 361 training samples and 40 testing samples are used. For the KLED-2019 dataset, 1325 training samples and 147 testing samples are used. We report the RMSE score averaged over 10-folds.

4.3 Baseline Computation

The baseline for the KLED-2020 dataset is determined by computing the average desktop luminance for each of the 3 user positions (VVF₁, VVF₂ and VVF₅) from their corresponding ceiling image. The ceiling-based average luminance (L_{ceil}) is then used to estimate the eye-level average luminance through $L_{eye} = \mathbf{a} \cdot L_{ceil} + \mathbf{b}$, where $\mathbf{a}$ and $\mathbf{b}$ are obtained from MODEL B as described by Kruisselbrink et al.[28]. The MODEL B is a linear model which is based on multiple luminance measurements for user positions in a mock-up office space. It uses the earlier mentioned formulation together with values for constants $\mathbf{a}$ and $\mathbf{b}$ which could be used to estimate luminance at a level of desktop, monitor, 40°luminance band (B40 luminance) and retinal [28]. The estimated average desktop luminance values are used as baselines to LUMNET predictions. Similarly, the baseline for the KLED-2019 dataset is determined by computing the ceiling-based average desktop luminance for both the VVF₂ and VVF₃ user positions from their corresponding ceiling image. The eye level luminance is then estimated with a calibration factor $\mathbf{k}$ as $L_{eye} = \mathbf{k} \cdot L_{ceil}$.

4.4 Results

We evaluate our approach for estimating the VVFDL using both supervised and self-supervised learning. In each experiment, LUMNET is trained with 10-fold cross-validation and the mean RMSE score over the 10-folds is reported along with the standard deviation. Cross-validation was only relevant for obtaining the results of LUMNET's performance, and not the baseline results, because the baseline results are fixed, as explained in Section 4.3. There

(a) KLED-2020

(b) KLED-2019

(c) Ceiling-based image from the KLED-2020 dataset.

(d) Ceiling-based image from the KLED-2019 dataset.

Fig. 7. Floor plan of the experimental setups as adopted for the collection of KLED-2020 and KLED-2019 datasets. The ceiling-based luminance cameras are indicated in red while blue is used to denote the vertical visual field of a user. For the KLED-2020 dataset (a, c), the ceiling-based luminance camera is placed above the aisle with an orientation of 20° relative to the ceiling and close to the VVF₁ and VVF₂ positions. For the KLED-2019 dataset (b, d), the ceiling-based luminance camera is placed directly above the desktop areas.

we are not reporting standard deviation values for the baseline results. In our supervised learning approach, LUMNET is trained on labeled instances from the KLED-2019 and KLED-2020 datasets, respectively. As shown in Table 2, the results indicate a superior performance of LUMNET over the baseline, with an overall improvement

Table 2. Comparison of LUMNET with domain-knowledge driven baseline for luminance estimation. RMSE scores of the proposed model are averaged over 10-folds along with their corresponding standard deviation values.

	RMSE (cd/m^2) per VVF position				
	KLED-2019		KLED-2020		
	VVF ₂	VVF ₃	VVF ₁	VVF ₂	VVF ₅
Baseline	26.48	51.23	318.71	21.15	17.46
LUMNET	7.74 $\pm$ 3.11	14.47 $\pm$ 3.37	135.62 $\pm$ 111.74	10.27 $\pm$ 6.60	16.34 $\pm$ 8.14

of 38.5% for the KLED-2020 and 71.3% for the KLED-2019. It indicates that our approach is able to estimate the VVFDL with a greater precision through directly learning from raw labeled data as compared to the existing domain-knowledge based approach. The results in Table 2 also show a high standard deviation from the mean for some VVF positions, such as the VVF₁ position of the KLED-2020, it could be due to the presence of outliers in the dataset, as also observed by Kruisselbrink et al [28].

Table 3. Evaluating Knowledge Transferability. We pre-train LUMNET on the KLED-2019 and fine-tune it on subsets of various sizes of KLED-2020 to determine how well the learned representations transfer across different environmental settings and percentage of labeled data.

	Training sample size (%)	RMSE (cd/m^2) per VVF position		
		VVF ₁	VVF ₂	VVF ₅
Baseline	100%	318.71	21.15	17.46
LUMNET	5%	447.38 $\pm$ 334.30	25.69 $\pm$ 16.12	46.81 $\pm$ 14.78
	10%	251.49 $\pm$ 202.31	17.12 $\pm$ 8.05	42.20 $\pm$ 17.79
	20%	209.15 $\pm$ 160.80	13.52 $\pm$ 6.74	29.67 $\pm$ 15.74
	40%	164.67 $\pm$ 157.46	12.03 $\pm$ 7.64	22.44 $\pm$ 12.68
	80%	188.27 $\pm$ 188.37	10.44 $\pm$ 10.47	14.81 $\pm$ 10.97
	100%	157.43 $\pm$ 147.01	7.76 $\pm$ 3.77	15.49 $\pm$ 8.87

We also examine LUMNET's ability to generalize when fine-tuned on out-of-domain (unseen) data, through leveraging transfer learning. LUMNET is pre-trained on the KLED-2019 dataset and it is fine-tuned on subsets of the KLED-2020, with a small learning rate of $5e^{-4}$. These subsets represent percentages of the training samples available for learning the model after splitting the dataset during cross-validation. It is important to note that these percentages are not applied to the test samples. In order to utilize earlier learned representation, we re-purpose the pre-trained model by replacing an existing output layer with a randomly initialized layer that contains 3 hidden units to produce luminance values corresponding to the VVF₁, VVF₂ and VVF₅ positions, respectively. We fine-tune the repurposed model with a small learning rate for 100 epochs, to ensure that the pre-trained features from the KLED-2019 are gradually adapted to the KLED-2020 and the weights are slowly updated. As shown in Table 3, the results demonstrate an improvement over the baseline, for subsets as small as 10% of the training samples, which represents just 36 ceiling-based images. It clearly highlights the effectiveness of LUMNET representations that they generalize well across dataset.

We verify the effectiveness of our self-supervised approach by evaluating the performance on adownstream task of lumniance estimation when LUMNET is pre-trained with unlabeled data using a contrastive objective.

Table 4. Assessing the quality of self-supervised representations learned with contrastive objective on unlabeled data. We quantify the performance of self-supervised models pre-trained on unlabeled instances of KLED-2019 and a linear regression is trained on-top of fixed network for KLED-2020. We compare contrastive model with a fully-supervised models trained directly on labeled in-domain data. We notice key improvements in RMSE score with a pre-trained model when few-labeled data is available.

Training sample size (%)		RMSE (cd/m^2) per VVF position					
		Self-supervised learning			Supervised learning		
		VVF ₁	VVF ₂	VVF ₅	VVF ₁	VVF ₂	VVF ₅
LUMNET	5%	401.93 ± 281.67	24.55 ± 15.80	38.86 ± 16.84	444.28 ± 277.45	28.74 ± 15.84	51.28 ± 14.95
	10%	224.08 ± 156.84	17.99 ± 12.93	41.95 ± 19.94	267.15 ± 141.26	21.96 ± 8.77	42.33 ± 16.63
	20%	174.68 ± 143.50	16.71 ± 11.86	33.29 ± 13.99	206.56 ± 117.77	17.84 ± 7.54	35.66 ± 12.64
	40%	193.54 ± 166.96	15.56 ± 10.95	29.56 ± 12.42	213.19 ± 163.97	15.06 ± 10.51	26.46 ± 11.51
	80%	177.05 ± 120.38	14.36 ± 10.03	24.78 ± 12.87	222.08 ± 168.65	10.79 ± 7.42	18.87 ± 10.60
	100%	179.26 ± 157.66	11.48 ± 9.91	23.84 ± 15.94	135.62 ± 111.74	10.27 ± 6.60	16.34 ± 8.14

LUMNET is pre-trained with ceiling-based images of the KLED-2019, without annotations, and the learned features are transferred onto labeled subsets of the KLED-2020. Similarly, in order to achieve this, we repurpose the pre-trained model through replacing the output layer of contrastive model with a randomly initialized layer that contains 3 hidden units to produce luminance values corresponding to the VVF₁, VVF₂ and VVF₅ positions, respectively. We freeze the convolution base (or encoder), that is, all the layers before a fully connected layer (see Figure 4) and train a linear regression layer on top of the frozen network with a learning rate of $5e^{-3}$, for 100 epochs. In this way, we use the pre-trained model as a fixed feature extraction mechanism, given that the utilized dataset is limited in size. As shown in Table 4, the results show an improvement over the baseline for larger subset sizes, such as 100% of the samples (361 samples), for the VVF₁ and VVF₂ positions. It indicates that LUMNET is able to estimate the VVFDL on the KLED-2020 with a reasonable performance through effectively leveraging self-supervised learning.

The results in Table 4 also highlight a better performance of LUMNET when trained with self-supervised learning, as compared to supervised learning. It is expected, given that our model learns meaningful representations from the KLED-2019 dataset, when solving the contrastive pretext task. Likewise, the learned features from the KLED-2019 are found to be highly generalizable when a linear layer is added on-top of LUMNET's encoder for learning to estimate the luminance on the KLED-2020, as the encoder or convolutional base layers are frozen. In this case, the self-supervised model improves the estimation as compared to their supervised counterparts for subsets as small as 5% of the training samples, which represents just 18 ceiling-based images. Concretely, if the ceiling-based images are collected with 10 minutes interval such as in [27, 28], a maximum of 3 hours would be needed for an adaptive lighting control system to capture 18 ceiling-based images and make use of the pre-trained LUMNET with our self-supervised approach which is several folds efficient than collecting and labeling large dataset. Furthermore, the performance of LUMNET is also evaluated when it is pre-trained on unlabeled examples from the KLED-2019 with contrastive learning, and trained on a downstream task of luminance estimation with labeled subsets of the same dataset. It is achieved through freezing the initial 2 convolution layers of the pre-trained model and fine-tuning the rest with a learning rate of $5e^{-4}$, for 100 epochs. As shown in Table 5, we see an improvement over the baseline for larger subset sizes, such as 100% of the samples (361 samples), and an improvement over the results of training LUMNET directly with labeled data. It is also expected, as contrastive models effectively learn general-purpose representations from unlabeled data.

Finally, given that the KLED-2020 dataset is limited in size, we examined the performance of LUMNET when trained with data augmentation. It is a crucial approach for effectively learning deep models and improving their generalization capability while reducing overfitting. Here, we set brightness values in the range [0.2, 1.0], where

Table 5. Generalization of the self-supervised learned features under low-data regime. The LUMNET is pre-trained on unlabeled instances of KLED-2019 with a contrastive loss and initial layers are frozen for learning downstream task of luminance estimation with varying percentages of labeled instances from the KLED-2019.

Training sample size (%)		RMSE (cd/m^2) per VVF position			
		Self-supervised learning		Supervised learning	
		VVF ₂	VVF ₃	VVF ₂	VVF ₃
LUMNET	5%	28.31 ± 9.76	69.19 ± 10.36	28.33 ± 10.28	72.70 ± 13.86
	10%	18.85 ± 6.62	38.50 ± 14.01	19.55 ± 6.68	45.78 ± 25.54
	20%	17.61 ± 7.45	35.07 ± 16.47	18.88 ± 6.01	43.73 ± 22.58
	40%	9.65 ± 3.27	16.19 ± 10.76	12.02 ± 4.47	19.54 ± 11.86
	80%	6.91 ± 1.49	12.41 ± 3.15	10.32 ± 4.04	15.43 ± 5.64
	100%	6.24 ± 1.81	10.27 ± 2.62	7.74 ± 3.11	14.47 ± 3.37

Table 6. Effectiveness of data augmentation for luminance estimation models on KLED-2020. We utilize four input transformations which includes brightness in a range of [0.2, 1.0], vertical and horizontal (V. & H.) shifts of 10% and 20%, as well as Gaussian blur with radius (r) 5 and 10.

Operation		RMSE (cd/m^2) per VVF position		
		VVF ₁	VVF ₂	VVF ₅
Baseline	—	318.71	21.15	17.46
LUMNET	Brightness	227.65 ± 201.17	18.25 ± 9.48	34.4 ± 19.70
	Shift (10%)	194.73 ± 142.08	16.86 ± 10.93	34.36 ± 15.18
	Shift (20%)	251.84 ± 184.64	15.97 ± 8.40	37.31 ± 22.26
	Blur (r = 05)	208.72 ± 215.52	14.96 ± 10.91	33.42 ± 24.12
	Blur (r = 10)	292.95 ± 280.94	17.79 ± 15.52	38.35 ± 27.41

these values represent percentages of the actual image brightness. Similarly, the Gaussian blur augmentation is also applied, with radius values of 5 and 10. We also utilize vertical and horizontal shift transformation, where the images are randomly shifted vertically and horizontally by 10% and 20% of their original height and width, respectively. The results are presented in Table 6. We observe an improvement over the baseline in each of these experiments, for the VVF₁ and VVF₂ positions, but no significant improvement over the results without augmentation of Table 2. It could be an indication that the augmented images do not exactly reflect the original images and severely drifts from the actual data distribution and other techniques like AutoAugment [8] could be utilized to learn data-specific augmentations that improves estimation.

5 DISCUSSION

It can be seen from the experimental results that LUMNET achieves a better performance than a domain-knowledge driven baseline, when it is trained directly on labeled samples of the KLED-2019 and KLED-2020 datasets, as indicated by the lower RMSE score. This also holds when LUMNET is pre-trained on either of the datasets and fine-tuned on large subsets of the training examples of the other dataset, using transfer learning. Here, we either freeze the entire convolution base of the pre-trained model and extract features from the penultimate layer as the

source and target domain datasets are similar and it reduces the risk of overfitting or we fine-tune the entire network end-to-end with a smaller learning rate which improves the performance even further because the weights are slowly updated. Moreover, we notice that utilizing transfer learning results in a faster convergence and better generalization while greatly reducing the time required for learning. We believe this is beneficial when deploying and using LUMNET in indoor spaces directly on-device. Similarly, our self-supervised contrastive learning approach provides a systematic way to incorporate unlabeled data in the learning process which is generally abundant in real-world. Our method achieves significant performance improvement with low-labeled data, which indicates that the contrastive network learns widely useful features. We note that it would be handy in practical situations where data annotation is expensive and time-consuming.

As with other deep learning tasks, a major challenge with the luminance estimation task is the low availability of labeled data, given that labeled data is tedious and costly to obtain in real-life scenarios. LUMNET tackles this using our SSL approach. Our SSL approach is able to achieve a better performance than the baseline because LUMNET learns valuable image representations when pre-trained with unlabeled data, using the approach described in Section 3.5. Another challenge with the luminance estimation task is to improve the performance by minimizing the RMSE score. We observe that training LUMNET with normalized luminance values, using the min-max normalization together with the binary cross-entropy loss, results in a better performance than the standard normalization or no normalization. Such performance is expected, given that the output is scaled within the range $[0, 1]$, and can become relatively easier for the network to quantify as compared to unscaled values of arbitrary range. It is also suitable for the sigmoid activation function, which is used in the output layer of the network. Furthermore, we notice that training LUMNET on the KLED-2020 dataset with data augmentation does not improve the performance. This is unexpected, given that data augmentation generally enables the model to generalize better and prevents overfitting. However, the low performance indicates that the augmented examples do not reflect the original samples and hence, negatively affect the learning process. It could also be due to the choice of the factors or ranges used in the augmentation methods. We believe that leveraging techniques that automatically discover optimal augmentation strategies during learning could be beneficial to discover useful augmentations and corresponding factors to improve generalization. Such a technique is proposed in AutoAugment [8], to automatically search and apply the best data augmentation operations, such as rotation and translation, on the images.

Low dynamic range (LDR) images such as PNG images, typically have a 24-bits encoding, 8-bits for each of the R, G, and B color channels [13]. This results in a range of $[0, 255]$ intensity levels for each channel of a pixel. However, this encoding is not suitable to contain the dynamic range of natural scenes in the real world, as this dynamic range would require far more than 256 intensity levels [13]. An HDR image of a scene is generated by fusing multiple LDR exposure photographs of the scene to capture its wide luminance range [18]. Our conversion of HDR images to PNG therefore results in some loss of data, given that the RGB channels of each pixel of the resulting PNG image are limited to integers in the range $[0, 255]$, unlike the floating point representation of HDR pixel values. While LUMNET is trained with labeled data including luminance values computed from the floating-point pixel values of the ceiling-based HDR images, the actual images which are passed as input to the model, are converted to PNG. We believe the neural network can achieve a better performance if trained with the original HDR images.

6 CONCLUSION

In this paper, we present LUMNET, a deep neural network for estimating desktop luminance based on deep convolutional neural network that can be incorporated in adaptive lighting control systems. LUMNET learns visual representations from ceiling-based images and predicts the average desktop luminance as experienced by the user in a real-world setting. Our approach achieves a better performance than prior methods based on

domain-knowledge in estimating luminance, when trained on labeled datasets, namely the KLED-2019 and KLED-2020. In particular, our experiments show an overall improvement of 38.5% for the KLED-2020 and 71.3% for the KLED-2019, on average, accross all VVF positions, as compared to baseline results. We also propose a self-supervised constrastive learning approach to learn image features from unlabeled data and use it in a downstream task with a transfer learning. We demonstrate that self-supervised pre-training significantly reduces the labeled data requirement and even with 20% annotated data our model achieves a RMSE of $44.97cd/m^2$ for the VVF₃ position of the KLED-2019 dataset, which is lower than the baseline using entire labeled data. Likewise, LUMNET's performance improves when augmented and non-augmented data are utilized together during the learning phase, and it helps the model to generalize well on out-of-domain data in transfer learning. Given that luminance metrics are closely related to human subjective preferences, the soundness of the results with LUMNET could be further verified with a user study. Therefore, for future work, we recommend a study in living or working spaces with users to compare the model performance in estimating their visual performance and adapting lighting to match human subjective experience. Furthermore, the LUMNET model can be deployed locally on a resource constrained device used to capture the ceiling-based images to ensure that users' privacy is preserved.

ACKNOWLEDGMENTS

This work is part of the Optilight project led by prof. dr. ir. Jean-Paul Linnartz. Optilight develops mathematical and analytical models for lighting optimizations and is jointly carried out by Eindhoven University of Technology and Signify. The main figure is adapted from undraw.co and various other icons used in figures are created by Yaroslav Samoilov, Sachin Modgekar, Pause08, Phonlaphat Thongsriphong, Vectorstall, Srinivas Agra from the noun project.

REFERENCES

- [1] Philip Bachman, R Devon Hjelm, and William Buchwalter. 2019. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems*. 15535–15545.
- [2] Ziad Boullos, Scott S Campbell, Alfred J Lewy, Michael Terman, Derk-Jan Dijk, and Charmane I Eastman. 1995. Light treatment for sleep disorders: consensus report: VII. Jet lag. *Journal of biological rhythms* 10, 2 (1995), 167–176.
- [3] PR Boyce. 1973. Age, illuminance, visual performance and preference. *Lighting Research & Technology* 5, 3 (1973), 125–144.
- [4] Peter R Boyce, Jennifer A Veitch, Guy R Newsham, CC Jones, J Heerwagen, M Myer, and CM Hunter. 2006. Occupant use of switching and dimming controls in offices. *Lighting Research & Technology* 38, 4 (2006), 358–376.
- [5] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. 2018. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 132–149.
- [6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709* (2020).
- [7] Antonia Creswell, Kai Arulkumaran, and Anil A Bharath. 2017. On denoising autoencoders trained to minimise binary cross-entropy. *arXiv preprint arXiv:1708.08487* (2017).
- [8] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. 2018. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501* (2018).
- [9] Christopher Cuttle. 2004. Brightness, lightness, and providing 'a preconceived appearance to the interior'. *Lighting Research & Technology* 36, 3 (2004), 201–214.
- [10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.
- [11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
- [12] LT Doulos, A Tsangrassoulis, CA Bouroussis, and FV Topalis. 2013. Reviewing drawbacks of conventional photosensors: are CCD/CMOS sensors the next generation?. In *Lux Europa 2013, 12th European Lighting Conference*.
- [13] Tao Fang. 2009. On performance of lossless compression for HDR image quantized in color space. *Signal Processing: Image Communication* 24, 5 (2009), 397–404.

- [14] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. 2019. Slowfast networks for video recognition. In *Proceedings of the IEEE international conference on computer vision*. 6202–6211.
- [15] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. 2018. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728* (2018).
- [16] Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, et al. 2016. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *Jama* 316, 22 (2016), 2402–2410.
- [17] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- [18] Mehlika N Inanici. 2006. Evaluation of high dynamic range photography as a luminance data acquisition system. *Lighting Research & Technology* 38, 2 (2006), 123–134.
- [19] Douglas A Kerr. 2010. The CIE XYZ and xyY color spaces. *Colorimetry* 1, 1 (2010), 1–16.
- [20] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [21] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).
- [22] Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems*. 3294–3302.
- [23] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. 2019. Revisiting self-supervised visual representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 1920–1929.
- [24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.
- [25] Thijs Willem Kruisselbrink. 2020. *Practical and continuous luminance distribution measurements for lighting quality*. Ph.D. Dissertation. Department of the Built Environment. Proefschrift.
- [26] Thijs Kruisselbrink, Myriam Aries, and Alexander Rosemann. 2017. A practical device for measuring the luminance distribution. *International Journal of Sustainable Lighting* 19, 1 (2017), 75–90.
- [27] Thijs Kruisselbrink, Rajendra Dangol, and Evert van Loenen. 2019. Ceiling-based luminance measurements: a feasible solution? (2019).
- [28] Thijs W Kruisselbrink, Rajendra Dangol, and Evert J van Loenen. 2020. Feasibility of ceiling-based luminance distribution measurements. *Building and Environment* 172 (2020), 106699.
- [29] Rikard Küller and Lennart Wetterberg. 1993. Melatonin, cortisol, EEG, ECG and subjective comfort in healthy humans: Impact of two fluorescent lamp types at two light intensities. *International Journal of Lighting Research and Technology* 25, 2 (1993), 71–80.
- [30] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. 1989. Backpropagation applied to handwritten zip code recognition. *Neural computation* 1, 4 (1989), 541–551.
- [31] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. 2017. Fully convolutional instance-aware semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2359–2367.
- [32] Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 2017. Knowing When to Look: Adaptive Attention via A Visual Sentinel for Image Captioning. *arXiv:cs.CV/1612.01887*
- [33] Olvi L Mangasarian and David R. Musicant. 2000. Robust linear and support vector regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 9 (2000), 950–955.
- [34] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *nature* 518, 7540 (2015), 529–533.
- [35] T Moore, DJ Carter, and AI Slater. 2002. A field study of occupant controlled lighting in offices. *Lighting Research & Technology* 34, 3 (2002), 191–202.
- [36] Ali Motamed, Laurent Deschamps, and Jean-louis Scartezzini. 2015. Validation and preliminary experiments of embedded discomfort glare assessment through a novel HDR vision sensor. In *Proceedings of International Conference CISBAT 2015 Future Buildings and Districts Sustainability from Nano to Urban Scale*. LESO-PB, EPFL, 235–240.
- [37] Ali Motamed, Laurent Deschamps, and Jean-Louis Scartezzini. 2017. On-site monitoring and subjective comfort assessment of a sun shadings and electric lighting controller based on novel High Dynamic Range vision sensors. *Energy and Buildings* 149 (2017), 58–72.
- [38] Hieu V Nguyen and Li Bai. 2010. Cosine similarity metric learning for face verification. In *Asian conference on computer vision*. Springer, 709–720.
- [39] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).
- [40] Taesung Park, Alexei A Efros, Richard Zhang, and Jun-Yan Zhu. 2020. Contrastive learning for unpaired image-to-image translation. *arXiv preprint arXiv:2007.15651* (2020).
- [41] Timo Partonen and Jouko Lönnqvist. 2000. Bright light improves vitality and alleviates distress in healthy people. *Journal of Affective disorders* 57, 1-3 (2000), 55–61.

- [42] S Patro and Kishore Kumar Sahu. 2015. Normalization: A preprocessing stage. *arXiv preprint arXiv:1503.06462* (2015).
- [43] Alec Radford, Luke Metz, and Soumith Chintala. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434* (2015).
- [44] Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, Matthew P. Lungren, and Andrew Y. Ng. 2017. CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. *arXiv:cs.CV/1711.05225*
- [45] Mark S Rea and Michael J Ouellette. 1991. Relative visual performance: A basis for application. *Lighting Research & Technology* 23, 3 (1991), 135–144.
- [46] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*. 91–99.
- [47] Aaqib Saeed, Victor Ungureanu, and Beat Gfeller. 2020. Sense and Learn: Self-Supervision for Omnipresent Sensors. *arXiv preprint arXiv:2009.13233* (2020).
- [48] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. 2013. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229* (2013).
- [49] Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, Sergey Levine, and Google Brain. 2018. Time-contrastive networks: Self-supervised learning from video. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1134–1141.
- [50] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. 2014. CNN features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 806–813.
- [51] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).
- [52] Robert H Spector. 1990. Visual fields. *Clinical methods: The history, physical, and laboratory examinations* (1990).
- [53] Aravind Srinivas, Michael Laskin, and Pieter Abbeel. 2020. CURL: Contrastive Unsupervised Representations for Reinforcement Learning. *arXiv:cs.LG/2004.04136*
- [54] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. 2018. A survey on deep transfer learning. In *International conference on artificial neural networks*. Springer, 270–279.
- [55] WJM Van Bommel and GJ Van den Beld. 2004. Lighting for work: a review of visual and biological effects. *Lighting research & technology* 36, 4 (2004), 255–266.
- [56] Kevin Van Den Wymelenberg and Mehlika Inanici. 2014. A critical investigation of common lighting design metrics for predicting human visual comfort in offices with daylight. *Leukos* 10, 3 (2014), 145–164.
- [57] Jennifer A Veitch and Guy R Newsham. 2000. Preferred luminous conditions in open-plan offices: Research and practice recommendations. *International Journal of Lighting Research and Technology* 32, 4 (2000), 199–212.
- [58] Jennifer A Veitch, Guy R Newsham, Peter R Boyce, and CC Jones. 2008. Lighting appraisal, well-being and performance in open-plan offices: A linked mechanisms approach. *Lighting Research & Technology* 40, 2 (2008), 133–151.
- [59] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3156–3164.
- [60] Xiaolong Wang and Abhinav Gupta. 2015. Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE international conference on computer vision*. 2794–2802.
- [61] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. 2017. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2097–2106.
- [62] Greg Ward. [n.d.]. *Anywhere Software*. <http://www.anywhere.com/>
- [63] Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. 2018. Unsupervised Feature Learning via Non-Parametric Instance-level Discrimination. *arXiv:cs.CV/1805.01978*
- [64] Richard Zhang, Phillip Isola, and Alexei A Efros. 2016. Colorful image colorization. In *European conference on computer vision*. Springer, 649–666.
- [65] Richard Zhang, Phillip Isola, and Alexei A Efros. 2017. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1058–1067.