IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received April 8, 2021, accepted May 11, 2021, date of publication May 19, 2021, date of current version May 27, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3081736

An Autonomous Illumination System for Vehicle Documentation Based on Deep Reinforcement Learning

LAMPROS LEONTARIS[1], (Member, IEEE), NIKOLAOS DIMITRIOU[1], (Member, IEEE), DIMOSTHENIS IOANNIDIS[1], KONSTANTINOS VOTIS[1], DIMITRIOS TZOVARAS[1], (Senior Member, IEEE), AND ELPINIKI PAPAGEORGIOU[2], (Senior Member, IEEE)

[1]Information Technologies Institute, Centre for Research and Technology Hellas, 57001 Thessaloniki, Greece
[2]Department of Energy Systems, Faculty of Technology, University of Thessaly at Geopolis, 41500 Larissa, Greece

Corresponding author: Lampros Leontaris (lleontar@iti.gr)

This work was supported in part by the European Commission through the Optimizing Manufacturing Processes through Artificial Intelligence and Virtualization (OPTIMAI) Project funded by the Horizon 2020 Research and Innovation Program under Agreement 958264, and in part by the European Union through the General Secretariat of Research and Technology, Ministry of Development and Investments by the Bilateral Science & Technology (S&T) Cooperation Program Greece—Germany (2017) under Project INVIVO/T2DGE-0951.

ABSTRACT A common problem for machine vision applications is uncontrolled illumination conditions that cause undesired artifacts on sensorial data. For instance, quality inspection using color cameras, while having wide industrial application, requires manual illumination adjustment and is severely affected by external lighting sources and the physical properties of the inspected object. To overcome this problem, we propose an autonomous illumination solution, that adjusts illumination via a Deep Reinforcement Learning (DRL) agent following a goal-oriented reward that takes into account image entropy and specularity. The system is validated in a challenging vehicle documentation use case where vehicle images are captured under various lighting conditions using a camera and an in-house built illumination system. The DRL agent learns to control illumination levels directly from high-dimensional visual inputs by mapping the interactions from the environment to the reward-driven control actions of the illumination system, targeting an optimal illumination zone even under the appearance of abrupt illumination changes in the environment.

INDEX TERMS Artificial intelligence, autonomous systems, computer vision, deep reinforcement learning, illumination control.

I. INTRODUCTION

Reinforcement Learning (RL) has substantially benefited from the recent achievements of Deep Learning (DL) methods which provide an efficient way to learn complex representations in various tasks including pattern recognition and computer vision. The recent advances in DL opened a research path for improving RL by deploying DL networks as complex functions approximators. Thus, efficient representations of high dimensional inputs can be successfully derived from Deep Reinforcement Learning (DRL) agents proving remarkable performance in benchmark problems [1]–[3]. However, the research in the application and implementation of RL in real-world problems has been mainly restricted by the difficulty of setting up the experimental environment which may in some cases be significantly complex or time demanding while the stability of RL in control tasks is considered an open research issue [4].

Illumination control is one of the most critical aspects for machine vision inspection applications where the inspection environment should be thoroughly analyzed in terms of reflections, multiple lighting sources, position of the lighting sources and equipment requirements [5]. Up to now, in industrial machine vision applications, illumination conditions are manually set up based on human experience, following a trial and error process. Automated solutions exists mainly for energy-saving in smart road lighting applications [6]–[8] or other applications that aim at optimizing lighting conditions in indoor environments [9]–[12]. In terms of lighting technologies, Light emitting diodes (LEDs) have been adopted

The associate editor coordinating the review of this manuscript and approving it for publication was Li Zhang.

in the recent years by various sectors including the industry, due to the considerable efficiency and performance over older lighting systems [13] and have been the main components of many smart lighting systems [14].

In our work, we provide an autonomous illumination system based on the DRL paradigm and guided by image quality. Existing research on learning-based methods for controlling and adjusting image quality, mainly focus on learning to expose, retouch or enhance photos [15]–[17] using image processing techniques including salient feature extraction which is a widely adopted methodology in various research areas [18]–[22]. Furthermore, most evaluations for these works are performed on benchmarks of scenery, human scenery or other image sceneries captured under fixed light sources such as natural external lighting or typical indoor light sources and aim at estimating or predicting illumination [23]–[25].

In automotive industry, documenting the condition of vehicle's surface under optimal ambient light is crucial for quality control since unsuitable lighting may lead to undesired reflections and shadows that make visual inspection tasks unreliable and inaccurate. Furthermore, in fleet management, monitoring and tracking the vehicles' surface under reliable lighting conditions is important for scheduling maintenance and ensuring roadworthiness. In this direction, our work proposes an autonomous system that adjusts illumination level during color images recordings of vehicles, based on predefined quality criteria. The system we propose comprises a vision sensor to capture the scene, a controllable illumination system to adjust the lighting conditions under which the scene is captured and a DRL agent deployed on a computation unit to control illumination. This work aims to bridge the gap in the literature between DRL methodologies and smart lighting control applications.

The main contributions of our work can be summarized as follows:

- We proposed a system architecture to explore the use of external illumination sources in a controlled scene, using LED luminaires and common low-cost electronic equipment.
- We investigated the control of image quality by adjusting the light projected on the scene. For this purpose, we defined an image quality metric in order to build a reward strategy for a DRL framework.
- We adjusted the Deep Q-Learning approach and explored the optimal set of parameters to achieve consistent results.

II. RELATED WORK

Methods based on DRL enable DL as functions approximators to scale to complex control tasks and provide efficient frameworks to build autonomous systems. DL frameworks have reached smart manufacturing providing advanced computational methods to improve manufacturing systems' performance [41] while demonstrating efficiency in monitoring industrial processes [42]–[46]. With the proliferation of DL methods there has been substantial progress in DRL with many successful applications in various domains including smart manufacturing and robotics, autonomous driving and energy.

Traditional tabular Q-learning approaches are used in experiments with low-dimensional states with the goal to learn the action-value function $Q(s, a)$ which measures the expected reward from taking any particular action at any state. Deep Q-Networks have been recently used to handle high dimensional inputs. In [47] authors introduce the potential of DRL application in smart grids while in [30], a goal-oriented application of DRL for the efficient scheduling optimization of electricity consumption in residential buildings is examined, using Deep Q-Learning and Deep Policy Gradient methods. An extension of DRL as a semi-supervised paradigm is proposed in [31], focusing on the problem of indoor localization in the Internet Of Things (IoT) and smart city applications, combining labeled and unlabeled data deploying a deep variational autoencoder for learning the best action policies. In the same domain, a DRL-based framework is proposed in [32] for building an energy management and scheduling agent with the goal of long-term energy efficiency in an IoT environment. A Deep Q-Network detection system is introduced in [33] to learn the optimal defending policy against data integrity attacks in smart grids and is evaluated for its detection accuracy and speed. A Double Deep Q-Learning network (DDQN) has been deployed in [34] to provide an efficient solution to the optimal active power dispatch problem. A recent work [48] examines the decision making potential of DRL in the autonomous IoT systems while in [49] the proposed DRL-based agent learns the optimal policy to deal with caching-based IoT data proving better performance in simulation results than the baseline policies.

In the autonomous-driving domain, authors in [50] explain the practical challenges in the autonomous-driving applications which are mostly built on simulated environments. In a recent work, [26], a Deep Q-Learning process is developed to find the optimal driving policy for the successful on-ramp merging for the automated driving systems (ADS) using a Long Short-Term Memory (LSTM) network to model the interactive environment. The performance of Deep Q-Network and Deep Deterministic Actor Critic algorithm has been examined in [27], for the two main categories of RL, discrete actions and continuous actions categories, evaluating the methods on a car simulator for the lane-keeping scenario in autonomous driving. In the same domain, a DRL agent, [28], for learning safe lane changing behavior and an attention mechanism is examined to boost the performance while the system is validated in a car simulator. In the domain of autonomous urban driving, the work of [29] proposes a Controllable Imitative Reinforcement Learning (CIRL) approach using Deep Deterministic Policy Gradient (DDPG) algorithm to achieve state of the art success in driving benchmark tests and generalization ability on diverse and unseen cases.

TABLE 1. Summary of related works for applications of deep reinforcement learning and smart lighting.

		Application Domain	Application Task	Reference	Methodology
	DRL applications	Autonomous Driving	On-ramp merge driving policy	[26]	Deep Q-Learning / Long Short-Term Memory
			Lane-keeping scenario	[27]	Deep Q-Learning / Deep Deterministic Actor Critic
			Lane-changing scenario	[28]	Attention-based Actor-Critic
			Diverse urban driving scenarios	[29]	Deep Deterministic Policy Gradient
		Smart grid/energy/city and autonomous IoT systems	Energy efficiency - optimization	[30]	Deep Q-Learning / Deep Policy Gradient
			Indoor localization using IoT data	[31]	semi-supervised DRL
			Energy management - scheduling	[32]	Deep Q-Learning
			Data integrity attacks - defending policy	[33]	Deep Q-Learning
			Optimal active power dispatch	[34]	Deep Q-Learning
		Smart manufacturing and robotics	Complex door opening skill learning	[35]	Deep Deterministic Policy Gradient
			Assembly skill learning	[36]	Deep Q-Learning
			High-precision assembly skill learning	[37]	Deep Q-Learning/ Long Short-Term Memory
			Job shop scheduling	[38]	Dueling Double Deep Q-Learning
			Manufacturing job scheduling	[39]	Multi-class Deep Q-Learning
			Semiconductor production scheduling	[40]	Deep Q-Learning
	Intelligent lighting control applications	Smart road lighting	Traffic flow-based light adjustment - energy efficiency	[6]	Sensor-based statistical decision making
			Energy efficiency - monitoring	[7]	Brute-force optimization
			Energy efficiency	[8]	Fuzzy-logic/ Neural network
		Indoor lighting	User activity -based light adjustment	[9]	Closed loop-device light control
			Energy efficiency- daylight adaptation	[10]	Multi-layer neural network
			Energy efficiency- daylight adaptation	[11]	Linear optimization

In smart manufacturing and robotics, frameworks based on RL have been studied to offer solutions to complex and hard to engineer behaviors [51], [52]. Minimizing human intervention and hard-engineering in enabling the learning of autonomous robotic systems has been an open goal in robotics, where manipulation skills are usually hard to train and require a level of supervision [35]. In [36], authors propose and validate a skill acquisition method based on DRL for industrial robotics, to provide a data-driven autonomous solution for the assembly process where the environmental uncertainty is a significant factor. For a similar task of skill acquisition of industrial robotics, authors in another work [37] deploy Recurrent Neural Networks (RNN) for training a DRL agent in order to learn to take the optimal actions by observing sensorial data from an 7-axis robot arm and provide an autonomous solution for a high precision fitting task. In manufacturing, recent research focuses on job scheduling frameworks to automate production and deploy a Dueling Double Deep Q-Network with prioritized replay to learn the best policy of actions for decision-making [38]. In [39], the authors adjust a multiclass DQN network for job shop scheduling problems on an edge computing framework while in [40] a semiconductor production scheduling case is studied by deploying multiple cooperative DQN agents for the optimization of the production in an autonomous manner. In order to create a taxonomy of DRL algorithms and intelligent illumination, we summarize the related works that focus on real-world applications of DRL, mainly for use cases of the industrial sector, in Table 1.

III. PROPOSED SYSTEM

A. ILLUMINATION SYSTEM AND DATA COLLECTION

In order to build a controlled environment in terms of the lighting conditions, we used two dimmable luminaires with LED (Light-Emitting Diode) [53] lighting technology with luminous power of 22,500 lumens each, in an environment where only the light emitted from the controlled lighting sources appears on the experimental scene. We selected the luminaires based on the high luminous power, the wide beam angle and the ability to be controlled with a DALI (Digital Addressable Lighting Interface) controller. DALI is an industry-standardized protocol specified in International Electrotechnical Commission (IEC) 62386 [54], [55] and is

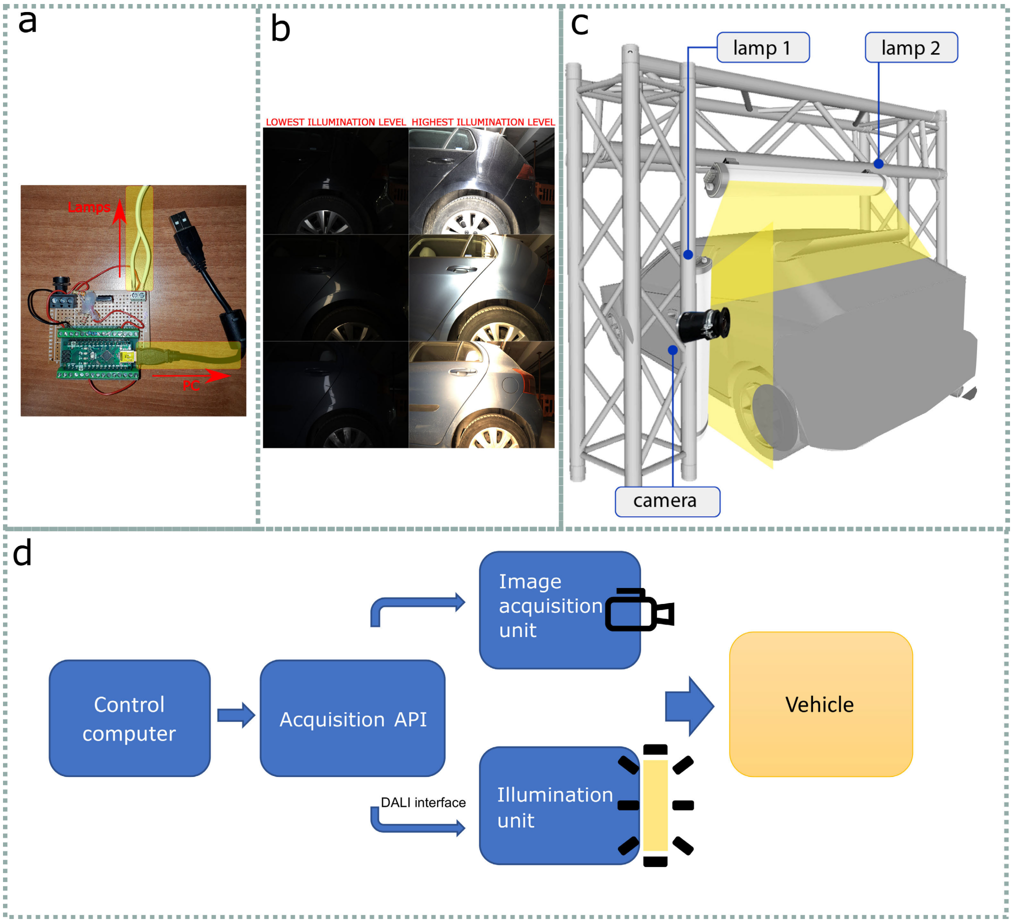

FIGURE 1. In (a), the in-house built electronic device which serves as a DALI communication bridge between the control PC and the luminaires. Indicative images captured under the minimum and maximum illumination levels from the luminaires are depicted in (b). On the right, in (c), we can see a schematic picture of the metallic structure that was used to mount the luminaires and the camera. In (d), the overall processing blocks of the system are illustrated.

commonly used for lighting control in smart illumination applications that require dimming simplicity and low light pulsation during brightness adjustment [56]. The developed illumination system comprises two LED luminaires and an in-house built DALI communication interface device, which was developed using an Arduino microcontroller. The communication device which connects the computer and the luminaires using DALI protocol are depicted in Figure 1a, where we see the endpoint wiring for the computer and the luminaires. Each of the luminaires has 255 possible dimming levels. The first 80 were not taken into consideration during the experiments since the difference between these levels was not significant. We defined 20 dimming levels $\{0, 80, 90, 100, 110, 120 \ldots 240, 250, 254\}$ for each lamp and we used the combination of the levels from the two luminaires which equals to 400 illumination levels. Indicative images captured under the lowest and highest setting of the illumination levels are depicted in Figure 1b.

The lighting system was placed in an underground indoor parking area where the external lighting sources are limited to the uneven illumination provided by fluorescent luminaires that are commonly used for lighting such areas. The main challenge in such environments, is the arbitrary and uneven distribution of ambient light emitted on the vehicle as captured from the camera’s viewpoint. This uncontrolled brightness combined with the complex geometry and high specularity of the vehicles’ surface causes undesired reflections and saturated regions on the images. The luminaires and the camera were mounted on a metallic structure underneath which the vehicle was parked in a fixed position. The metallic structure functions as a support frame for the luminaires and the camera and its modular construction allows for more luminaires or cameras to be added to the system in different positions. A schematic picture of the experimental setup is illustrated in Figure 1c, where the two luminaires are shown, mounted on the vertical and horizontal metallic pillars so that we can simulate light emitted from two perpendicular directions, from the side and from above. As can be seen in Figure 1d, the environment of the vehicle is captured visually by the camera sensor while the illumination module is used to act and change the lighting conditions of the environment.

During data collection, we used the DALI control interface to digitally control the luminaires from the computation unit using an Application Programming Interface (API) [57] based on open-access Arduino library [58]. The commands implemented in the API included increasing, decreasing and setting a fixed illumination level for the luminaires. For each luminaire, a total of 20 illumination levels were used during data collection so 400 images from the combinations of the illumination levels from the two luminaires were acquired. To augment the dataset, 3 images for each combination level were captured, collecting in this way 1200 images for each car. The ambient light from the environment was minimized by turning off the parking's lights, so that we can examine the effect of only the controllable light sources and safely assume that the quality of the picture, in terms of external lighting conditions, is only affected by the lighting system.

B. NETWORK ARCHITECTURE AND TRAINING PROCEDURE

The experimental setup was designed to investigate the application of DRL using the illumination system to provide an autonomous system that is optimized by experience-based learning during interaction with the environment based on Q-Learning method.

1) PROPOSED QUALITY METRIC AND REWARD STRATEGY

The first most crucial step for training a RL agent is to define the way that each new state, which in this work is observed by the image captured by the camera, returns a reward signal. For this purpose, we defined an image quality metric that incorporates image's entropy information as the first component and luminance intensity as the second component. The entropy component provides an objective statistical measure of the average information content that can be extracted from the image and has been examined in literature for assessing perceptual quality of images [59]. The luminance intensity is used to provide the information of the effect of the external lighting in the captured images.

The entropy information is extracted by the image histogram for each change of the external illumination level and the luminance intensity level is calculated directly from pixels with high intensity values in the 'L*' lightness component after converting the RGB (Red, Green, Blue) image to the CIELAB color space which is a common color system in industrial vision, closest to human perception [60], [61]. Converting the images from RGB to CIELAB color space provides a different color space where color is expressed by three values, the Lightness value 'L*', and 'a*', 'b*' that combine the colors red, green, blue and yellow. For our experiments, the separate component for lightness is important since we investigate the images in terms of illumination conditions. For the image entropy we used standard entropy equation defined as

$$E = \sum_{i=0}^{n-1} p_i \log_2(p_i) \tag{1}$$

where p are the histogram counts and n corresponds to the 256 levels of 8-bit image. The luminance component is calculated as

$$L = \frac{1}{P} \sum_{p}^{P} \mathbb{1}_{l(p) \geq thres} \tag{2}$$

where $\mathbb{1}$ is the indicator function of the set of pixels whose luminance intensity level is above the threshold *thres*, P is the number of pixels, $l(p)$ is the luminance intensity (L*) of pixel p. The luminance intensity (L*) is normalized in range [0, 255] and the threshold we used for our experiments was the value of 230 which was perceived as a saturation point by user experience.

In Figure 2d, we see the entropy component of the equation as observed in the images from the dataset in the order of increasing illumination level from the luminaires. The value of the image entropy is increasing as more light is emitted on the scene from the illumination system and more information from the image can be extracted. In Figure 2b, we see the luminance component of the equation which as expected illustrates the increasing trend of intensity for images captured under increasing illumination levels.

In order to incorporate perceptual criteria for the image quality we roughly approximate user experience by the feedback of 10 users that selected the illumination levels that were perceived as optimal. Based on this, we weight the luminance intensity component by a weight function representing the normal distribution from user's feedback. Thus, images acquired under very low or very high illumination levels contribute less to the luminance component. In Figure 2c, we see the estimated normal distribution from the users', regarding the illumination levels of images that are perceived to be better in terms of brightness. The normal distribution, $N(\mu, \sigma^2)$ is estimated with mean $\mu = 205.3$ and standard deviation as $\sigma = 20.2$. We can observe that the mean of the distribution corresponds to values close to the combined illumination level '200' while lower or higher levels are given lower weights. So if f_N is the probability density function of N, and E_i is the level of the external illumination system for image i, the weighted luminance component (2) is calculated as

$$L_w = f_N(E_i)L_i. \tag{3}$$

Moreover, we introduce weights to the entropy and luminance components, giving more influence to the information from the luminance component and the equation we used for the image quality measure is defined from (1), (3) as

$$q = w_1 E + w_2 L_w + c \tag{4}$$

where c is a shifting constant so that we can define two regions in the quality function $q \geq 0$ and $q < 0$ and

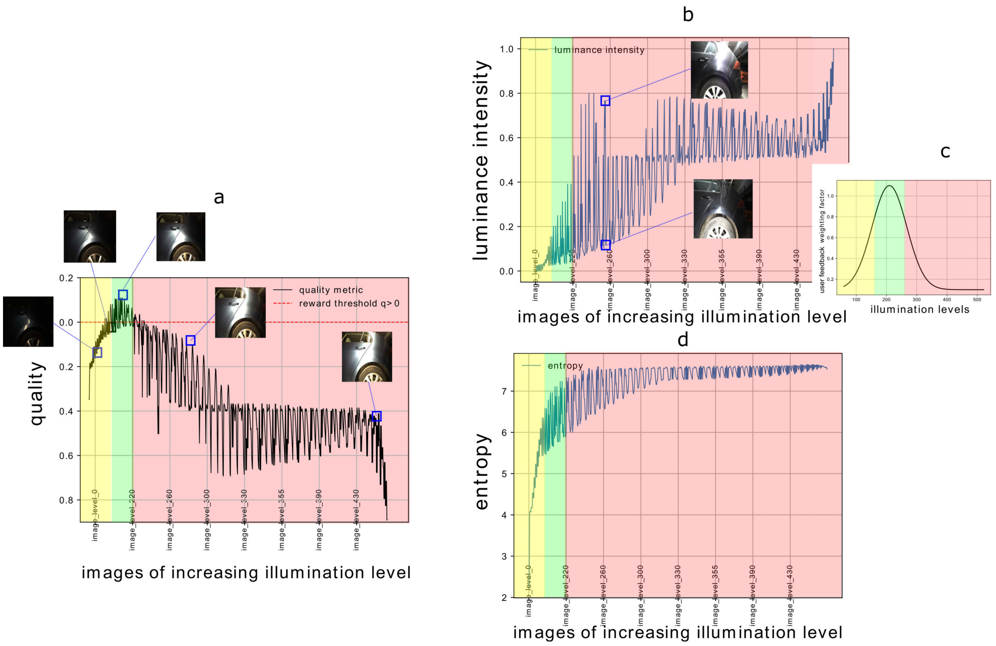

FIGURE 2. The defined quality metric (a) combines information from image's entropy (d) and luminance intensity values (b). (c) A rough estimation of a weighted normal function that provides weight factors for each illumination level obtained after asking 10 users to select in a qualitative manner which images are perceived as better in terms of brightness.

$w_1\ E^t + w_2\ L_w^t\ +c < 0$	$w_1\ E^t + w_2\ L_w^t\ +c \geq 0$
$r_t\ = 0$	$r_t\ = +1$

FIGURE 3. Reward strategy followed for time step *t* giving positive rewards only to states that the quality metric is calculated greater than zero.

w_1, w_2 the weights of the components. For our experiments, we qualitatively selected the values 0.3, 0.7 for w_1, w_2 to produce a weighted average between entropy and luminance intensity. Finally, we define our reward function based on the quality metric equation as

$$reward = \begin{cases} 1, & \textit{if}\ q \geq 0 \\ 0, & \textit{otherwise} \end{cases} \tag{5}$$

giving only positive rewards to images for which the quality metric is $q \geq 0$ as depicted in Figure 3. The final calculated quality metric can be observed in Figure 2a where the diagram plot of the quality metric against images of sorted illumination levels is depicted. The quality threshold $q \geq 0$ which is used to assign the reward-score to the image, is marked in dashed red line. Marked in the yellow region of the Figure 2b and 2d are images captured under very low illumination levels, for which both the entropy and the luminance intensity are very low. On the red regions of the graphs, the entropy is increasing and finally converging, while the luminance intensity reaches saturation levels.

The oscillations that can be noticed in the diagrams of entropy (Figure 2b) and luminance (Figure 2d) are due to their non-linear relation with illumination levels. This non-linearity is caused by the complex geometry of the vehicle and the two different illumination sources (Figure 1c). Concretely, for the same cumulative illumination, different illumination setups on the lamp have a very different effect in terms of reflections and specularities. Each of the luminaires emit light that is distributed on surfaces of the vehicles with different specularity such as the wheels and the windows. As an example, in Figure 2b, two images are depicted, captured with the same cumulative illumination level but having a different illumination effect on luminance. Therefore, a cumulative illumination don't accurately reflect the contribution of the illumination levels of each lamp to the total light emitted on the image. Nonetheless, this assumption significantly simplified problem formulation, while experiments showed that it didn't affect the convergence of the network.

For the green zone in Figure 2a, combining information from the two components as described from (4), the region above the threshold corresponds to images with high entropy values yet illumination levels that don't cause severe reflections and saturation. Indicative images from different illumination levels are illustrated in the figure to show how the reward strategy follows the quality metric criteria to benefit images close to the values as defined by the components of the quality metric's equation. Thus, the users' feedback provides a contribution to the reward signal which is used for the training phase.

With the proposed quality metric, we adequately model the perceptual effect of luminance intensity. While the proposed metric is only a rough approximation of human perception on image quality, it allows us to define an efficient reward strategy for our DRL agent.

2) DEEP Q-LEARNING APPROACH

In our experiments, we adopt Deep Q-Learning approach [1] which uses a DL network, in our case a 2D Convolutional Neural Network (CNN), to approximate the Q action-value function given color images, representing the states of the RL environment. Compared to the Q-Learning algorithm, which can be used to estimate the values of the state-action pairs in a tabular manner, using a DL network to approximate the Q function provides the ability to the agent to handle high-dimensional inputs which in our case are color images. The environment which consists of the current image of the vehicle, is fully observable to the DRL agent and each state-image is an image captured under a certain illumination level without incorporating any movement either from the lighting system or the camera. Each transition from one state to the next depends only on the current state and action, which is a set of illumination control actions, and does not depend on past transitions. Additionally, our problem regards the continuous task of repeatedly adjusting the lighting conditions of the environment without a pre-defined terminal state and therefore the process can be formulated as an infinite horizon Markov Decision Process (MDP).

The MDP of our task is defined by the tuple (S, A, P, γ, R) where S is a set of states of images I, $S = \{I_i\}$, A is the set of actions $A = \{increase, decrease, no\,action\}$, P is the state transition probability from the previous state to the next, γ is the discount reward factor and R is the reward function. To solve the underlying MDP, we use model-free RL algorithm based on Q-learning which does not require to build the explicit model of the environment using the P transition probability and we deploy Deep Learning to handle the high-dimensional visual inputs. Thus, we aim at searching for a policy that optimizes a performance criterion defined as γ-discounted criterion [62]. The policy aims at maximizing the expected cumulative sum of rewards, $E[r_0+\gamma r_1+\gamma^2\ r_2+ \cdots\gamma^3\ r_t+\cdots|s_0]$, where s_0 is an initial state and r_t the reward at time step t.

The two main techniques used in Deep Q-Learning Network (DQN) that address the instability problems, due to the strong temporal correlations that usually appear when training the agent, are the experience replay memory and the target network. Experience replay memory is a circular memory buffer that stores the transitions for each step that an action is taken and a new state is observed from the environment. The target network plays the role of the fixed network so that the Temporal-Difference (TD) error is calculated on a fixed target. The weights of the target network are updated at regular intervals to match the weights of the on-line policy network [1]. In Procedure 1, the training procedure using DQN algorithm and a user defined image quality metric is analyzed. The algorithmic steps concerning DQN algorithm are adopted from the main algorithm [1] and are not explicitly analyzed but are written for completeness to explain how we adjust the main algorithmic steps with a user-defined reward strategy designed for our experiments. The parameters including the experience memory size, the weight factors, the number of episodes and steps and threshold for the luminance intensity are selected through experimentation with different setups in order to find the optimal configuration. These setups for the DQN parameters were explored through grid search, considering for the experience memory size the values {800, 1000, 10000}, for the number of episodes

Procedure 1 Training Process Using DQN Algorithm and a User-Defined Image Quality Metric

```
Input: Vehicle color images under different illumination
       conditions
$R_{size} = 1000$, $num\_episodes = 1000$, $num\_steps =$
$500$, $\gamma = 0.99$;
Initialize networks $targetNet(x; \theta_{target})$,
$policyNet(x; \theta_{policy})$;
Initialize experience replay memory $R$;
for $episode \leftarrow 1$ to $num\_episodes$ do
    Reset environment;
    for $step \leftarrow 1$ to $num\_steps$ do
        select action $a_{step}$ with $\epsilon - greedy$ strategy;
        act on illumination level based on $a_{step}$:
        $\{increase, decrease, no\ action\}$;
        calculate image quality metric using (4);
        compute reward using (5);
        store transition to $R$ memory buffer;
        sample a batch from experience memory:
        $B = \{R_i\}_{i=1}^{K}$;
        foreach $(s, a, s', r) \in B$ do
            calculate temporal difference error using
            Bellman equation $\delta =$
            $Q^{policyNet}(s, a) - (r + \gamma \max_a Q^{targetNet}(s', a))$;
            calculate Huber Loss
            $L(\delta) = \begin{cases} \frac{1}{2}\delta^2, & if\ |\delta| \leq 1 \\ |\delta| - \frac{1}{2}, & otherwise \end{cases}$
        minimize $\frac{1}{|B|} \sum_{b \in B} L_b(\delta)$;
```

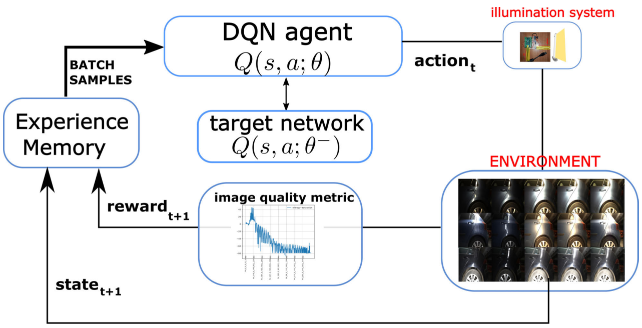

FIGURE 4. Training strategy of the DRL agent using experience memory technique and a target network for the optimization.

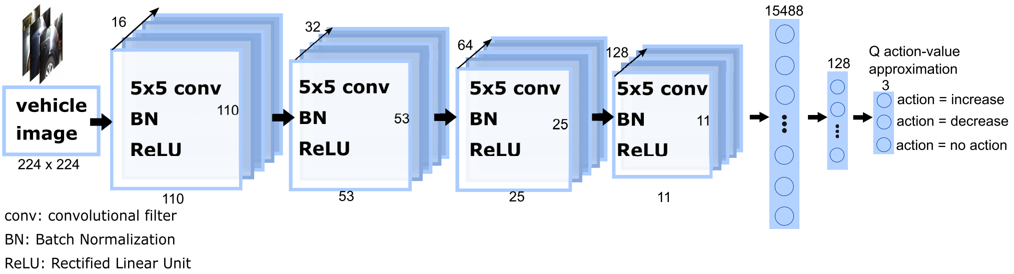

FIGURE 5. The network architecture used to approximate the Q action-value function for the DQN agent for the actions that control the illumination namely 'increase', 'decrease', 'no action' . The 2D-CNN receives color images as input which are propagated to the network's layers and the final outputs of the network are the three nodes that approximate the Q action-value for each of the three actions.

TABLE 2. Optimal parameters for the CNN network and DQN training procedure.

Parameter	Value
CNN #Layers	5
CNN #FC nodes	128
CNN #Filters per layer	16,32,64,128
DQN Experience memory size	1000
DQN #Episodes	1000
DQN #Steps	500
DQN gamma	0.99

the values {200, 500, 1000}, for the number of steps the values {100, 500} and for the gamma parameter the values {0.97, 0.98, 0.99}. The optimal values for the parameters are depicted in Table 2.

3) ARCHITECTURE OF THE PROPOSED AUTONOMOUS ILLUMINATION SYSTEM

The schematic diagram of the training strategy that was followed is depicted in Figure 4. For each new state of the environment which is represented by an image captured under certain illumination level, a reward-score is assigned which is used by the agent to update the network's parameters and generate Q action values approximations. The target network's parameters, are only updated after a defined number of steps to match the Q-network parameters, $\theta^{-} := \theta$.

For each new observation, a reward-score is assigned to the image based on the image quality metric. Each of these interactions with the environment are stored in the experience memory which is used to train the DQN agent. The DQN algorithm receives random samples from the experience memory and uses the target network for the optimization whose weights are normally frozen and is regularly updated with the parameters of the main policy network of the agent.

For our experiments, the target network was updated every 10 episodes which was found to be the best strategy for the stability of the training. Since the agent is trained on visual inputs using images from the cars, a 2D-CNN architecture, depicted in Figure 5, was used to approximate the Q action values. The network receives color images of 224×224 width and height from the cars and consists of four convolutional layers followed by one fully-connected layer (FC). The outputs generated by the network are approximated Q action values for each of the actions defined for the experiments which are *increase, decrease, no action*.

The design of the network, in terms of the number of convolutional layers, fully connected layers and network's parameters, was chosen by multiple experiments in order to have the best trade-off between computation time and results' quality. For this purpose, we used grid search to explore the combination of network hyper-parameters and we monitored the learning process using the reward scores achieved to conclude the optimal combination. For the number of convolutional layers we considered the values {3, 4, 5, 6}, for the number of filters for each layer the values {16, 32, 64, 128, 256} and for the number of the nodes of the fully connected layer the values {64, 128, 256}. The optimal combination of the network's hyper-parameters are depicted in Table 2. The computer resources used for the experiments include 64GB RAM memory, Nvidia GTX 1080Ti 11GB GPU and Intel i7-8700K processor and the computing time for training the agent was 8 hours and 50 minutes.

IV. RESULTS

To evaluate the proposed system, we used a total of 3600 images collected from three cars and we randomly split the dataset into train and test subsets in 2:1 ratio, resulting in 2400 images for train and 1200 images for test.

For the training phase, the algorithm was executed for 1000 episodes and for a maximum of 500 steps for each episode or terminated earlier if the agent selects an action that represent an inapplicable transition. The number of episodes and number of steps were selected with experimentation to achieve the best rewards. For each episode, the normalized score, which represents the average rewards achieved during the number of steps, is used to monitor the performance of the training through the episodes and is depicted in Figure 6.

For the testing phase, we used the test dataset of 400 images for each car which represented the testing environment. The trained agent was run for 2000 iterations so that we can examine the oscillations between the transitions. To test the stability we insert at regular intervals images with very

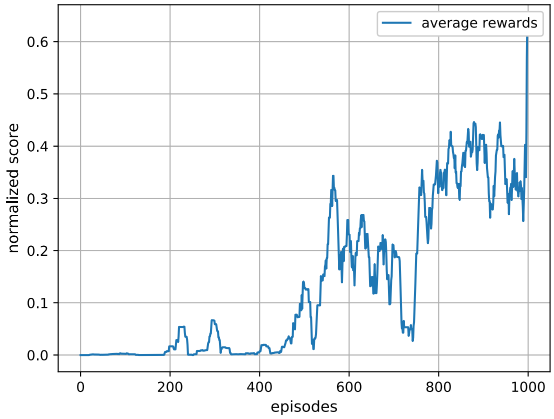

FIGURE 6. Average rewards monitored during the training phase for each episode and for the number of steps used. We notice the trend of increasing average of the rewards received through the episodes.

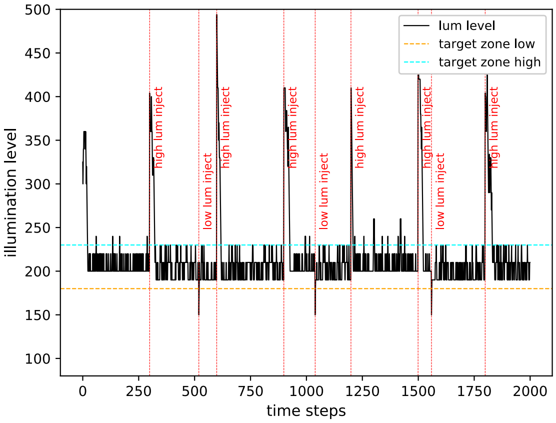

FIGURE 7. Time plot for 2000 iterations of the DRL agent acting on the environment's lighting changes for car 'A' while maintaining the stability of selecting the right action to remain inside in the target illumination zone.

high or very low illumination levels simulating an abrupt change in the environment and examine the agent's behavior. In Figure 7, this simulated process is depicted where we see the plot of the illumination level of the image as observed by the environment after the agent selects an action.

Concretely, starting from an image with a random illumination level at time step '0', the agent receives a random image from the dataset as the initial state of the environment, selects an action and the environment generates the next state which is an image with increased, decreased or same illumination based on the agent's action. Then, the agent receives the next state and selects an action, a process which is repeated for 2000 iterations. The target zone indicates the range between the illumination levels as defined by the user-feedback to have better image quality in terms of lighting conditions. The red vertical lines in the plot denote the time instants where a simulated abrupt illumination change is forced during the interaction with the environment. After these instants, the agent select actions that lead to the target zone with low oscillations between transitions. We notice that based on the visual input, the agent selects actions that control the illumination system so that the brightness of the environment is adjusted to optimal levels.

The simulation results for the other cars are depicted in Figure 8 where the agent, similarly to car 'A', efficiently selects actions for the illumination system that restores the illumination to levels defined inside the optimal target illumination zone. During the evaluation phase, the illumination step that is added/subtracted to the current illumination level is fixed and corresponds to the effect of the actions *increase,decrease*.

The number of transitions that are required to restore abrupt illumination changes to the optimal levels, depend on the illumination step used for the experiments, which is the change in the dimming level of the luminaires. Selecting larger or smaller step, that leads to faster or slower convergence to the target, depends on the application design requirements. For our architecture, instability issues appeared when experimenting with larger or smaller steps.

In Figure 9, we have a closer look at the time steps around an abrupt change that was inserted during simulation for the three cars. Each action indicator corresponds to the action the agent selects after observing the current image from the environment. We notice that after approximately 10 transitions from the 585 time step, after which the selected actions corresponds to 'decrease' the level, the agent is able to restore the illumination level to the target illumination levels defined in the target zone. Each of the transitions corresponds to a fixed step in the change of the illumination level, the same step defined during the training phase. The required time that the lighting system needs to change the illumination level, added to the time that the agent needs to receive an input and generates an output, is approximately 0.5 seconds. Thus, the 10 transitions after which the system converges to the target zone, corresponds to approximately 5 seconds. When a very high illumination change appears, which can be considered as an abrupt change in the lightness of the environment, the agent selects actions *decrease* ('-1') which decrease the illumination levels of the luminaires by the defined fixed step. After the next 10 steps approximately, during which the illumination levels are decreasing, the brightness of the images decreases as an effect of the illumination decrease from the luminaires. The agent converges to selecting action *no action* after reaching the target zone.

V. DISCUSSION AND LIMITATIONS

The number of transitions as can be visualized in Figure 9 corresponds to approximately 5 seconds duration which includes the time needed for the luminaires to increase their illumination level and the time needed for the trained agent's model to produce the output, times the 10 transitions required to reach stability levels. It should be noticed, that the response time of the luminaires consumes approximately 4 out of 5 seconds of the time needed for restoring the levels while the lightweight network architecture we used consumes the least time. This delay time does not match the typical response time of the

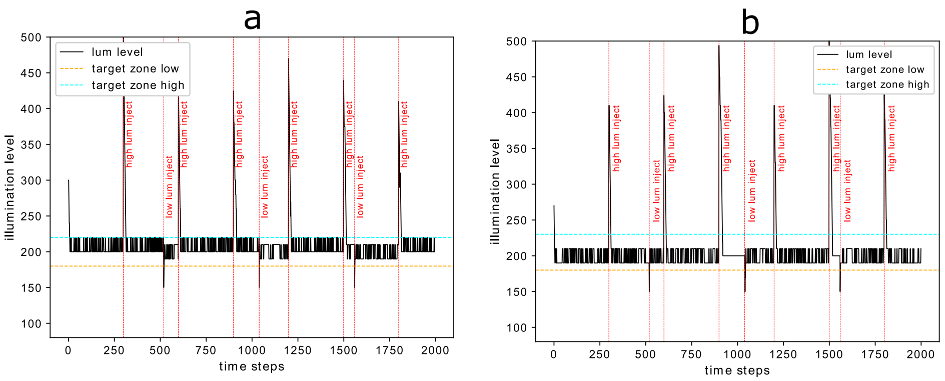

FIGURE 8. Time plots of the DRL agent acting on the illumination system to change the environment's lighting for car 'B' and car 'C' in (a) and (b) respectively. Notice that the agent learns to change the illumination correctly to match the image quality criteria set during the training phase.

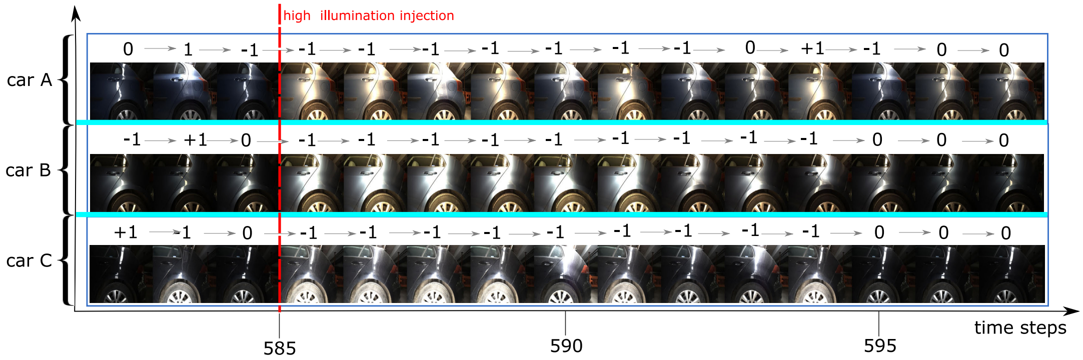

FIGURE 9. Indicative example focusing on the time steps 583-597 where we examine the behavior of the agent when an abrupt change in the lighting conditions appear. The agent receives the image from the camera as the current state and selects an action which in the figure is indicated as '+1' for increase, '−1' for decrease and '0' as no action.

DALI system, as a result of the suboptimal hardware implementation, since our solution in terms of physical hardware development opted for a low-cost and easy to build hardware implementation using common electronic components.

Thus, further improvement can be made on the proposed architecture, aiming at time-optimal adjustments, by using better physical interface and deploying different electronic design and programming. Another limitation of this work is the size of the dataset used, which should be further expanded, with more vehicles of different colors and surfaces, captured by multiple views, so that a more generalized solution is examined. Moreover, the quality metric that we defined, considers only entropy and luminance indices as subjective criteria of the quality of the image, which is suitable for the scope of this work but is limited for other use cases because it does not incorporate other factors such as contrast and texture which is commonly examined in image quality assessment studies [63].

Moreover, further experiments should be carried out to investigate the proposed system on a larger dataset using different image quality methodologies and different views from the vehicles so that to explore the potential of the system for quality inspection in automotive industry applications.

Our proposed quality metric approach can be compared to a recent work [64] where the authors studied a defect inspection case and defined an image quality assessment score by combining three indices, image visibility, image visibility distribution and overexposure, assigning to each one a defined weighting coefficient. The goal of their study is to adjust the brightness of a lighting source and examine how the accuracy of deep learning models for defect inspection is improved, by improving the quality of the images. The results of their study proved that the performance of the Deep Learning model was significantly improved for images with higher quality score, as defined by the authors. The performance was measured with F2-score showing a performance increase, when the worst and best quality images were used, from 47% to 80%. Compared to this study, in our work we focus on examining more thoroughly the adjustment of the image quality by providing an autonomous illumination system that is controlled directly by visual input signals from the images using DRL. For this purpose, we defined a different quality score based on the entropy and luminance of the images but also guided by human opinion feedback.

Therefore, the main outcomes of the conducted experiments can be summarized as follows:

- We demonstrated an application of DRL in controlling an illumination system for vehicle documentation based on a user-defined quality metric.
- The trained DRL agent was evaluated on the stability to choose the correct actions that lead to the target illumination zone set by the defined quality metric while showing the ability to cope with abrupt illumination changes in the environment.
- The lightweight 2D-CNN model that was used consume the least of the total response time required for restoring the stability levels which reflects the potential to improve the physical interface of our system in a time-optimal manner in order to minimize the system's total response time.

VI. CONCLUSION

In this work, we proposed an autonomous learning-based system to control an illumination system, adopting the Deep Q-Learning algorithm by introducing a user-defined image quality metric to build a reward strategy based on information extracted from image entropy and luminance intensity. The illumination system was developed using LED luminaires and in-house built components for the communication with a programming interface. The developed DRL agent is able to efficiently adjust illumination levels of the system to meet the image quality criteria set by the reward strategy. We tested the agent to run for a number of iterations, simulating a real-time process, showing stability even when abrupt illumination changes appear in the environment. The basic limitation of our work was the small dataset we used from three cars and from a single view from each car. It is expected that capturing images from a large number of vehicles, from different viewpoints, would require a different network architecture design to compensate for the increased complexity, created by the diverse specularities from the different parts of the vehicle.

ACKNOWLEDGMENT

This article reflects only the authors' view and the Commission is not responsible for any use that may be made of the information it contains.

REFERENCES

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[2] Y. Duan, X. Chen, R. Houthooft, J. Schulman, and P. Abbeel, "Benchmarking deep reinforcement learning for continuous control," in *Proc. 33rd Int. Conf. Int. Conf. Mach. Learn. (ICML)*, vol. 48, 2016, pp. 1329–1338.

[3] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," 2013, *arXiv:1312.5602*. [Online]. Available: http://arxiv.org/abs/1312.5602

[4] L. Buşoniu, T. de Bruin, D. Tolić, J. Kober, and I. Palunko, "Reinforcement learning for control: Performance, stability, and deep approximators," *Annu. Rev. Control*, vol. 46, pp. 8–28, Jan. 2018.

[5] Z. Liu, H. Ukida, P. Ramuhalli, and K. Niel, Eds., *Integrated Imaging and Vision Techniques for Industrial Inspection*. London, U.K.: Springer, 2015.

[6] G. Shahzad, H. Yang, A. W. Ahmad, and C. Lee, "Energy-efficient intelligent street lighting system using traffic-adaptive control," *IEEE Sensors J.*, vol. 16, no. 13, pp. 5397–5405, Jul. 2016.

[7] M. Mahoor, F. R. Salmasi, and T. A. Najafabadi, "A hierarchical smart street lighting system with brute-force energy optimization," *IEEE Sensors J.*, vol. 17, no. 9, pp. 2871–2879, May 2017.

[8] P. Mohandas, J. S. A. Dhanaraj, and X.-Z. Gao, "Artificial neural network based smart and energy efficient street lighting system: A case study for residential area in hosur," *Sustain. Cities Soc.*, vol. 48, Jul. 2019, Art. no. 101499.

[9] M.-S. Pan, L.-W. Yeh, Y.-A. Chen, Y.-H. Lin, and Y.-C. Tseng, "A WSN-based intelligent light control system considering user activities and profiles," *IEEE Sensors J.*, vol. 8, no. 10, pp. 1710–1721, Oct. 2008.

[10] A. Seyedolhosseini, N. Masoumi, M. Modarressi, and N. Karimian, "Daylight adaptive smart indoor lighting control method using artificial neural networks," *J. Building Eng.*, vol. 29, May 2020, Art. no. 101141.

[11] S. Borile, A. Pandharipande, D. Caicedo, L. Schenato, and A. Cenedese, "A data-driven daylight estimation approach to lighting control," *IEEE Access*, vol. 5, pp. 21461–21471, 2017.

[12] G. Boscarino and M. Moallem, "Daylighting control and simulation for LED-based energy-efficient lighting systems," *IEEE Trans. Ind. Informat.*, vol. 12, no. 1, pp. 301–309, Feb. 2016.

[13] C. Perdahci, H. C. Akin, and O. Cekic, "A comparative study of fluorescent and LED lighting in industrial facilities," *IOP Conf. Ser., Earth Environ. Sci.*, vol. 154, May 2018, Art. no. 012010.

[14] I. Chew, V. Kalavally, C. P. Tan, and J. Parkkinen, "A spectrally tunable smart LED lighting system with closed-loop control," *IEEE Sensors J.*, vol. 16, no. 11, pp. 4452–4459, Jun. 2016.

[15] J. Park, J.-Y. Lee, D. Yoo, and I. S. Kweon, "Distort-and-recover: Color enhancement using deep reinforcement learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5928–5936.

[16] Y. Hu, H. He, C. Xu, B. Wang, and S. Lin, "Exposure," *ACM Trans. Graph.*, vol. 37, no. 2, pp. 1–17, Jul. 2018.

[17] T. Vu, C. V. Nguyen, T. X. Pham, T. M. Luu, and C. D. Yoo, *Fast and Efficient Image Quality Enhancement via Desubpixel Convolutional Neural Networks* (Lecture Notes in Computer Science). Cham, Switzerland: Springer, 2019, pp. 243–259.

[18] G. Stavropoulos, P. Moschonas, K. Moustakas, D. Tzovaras, and M. G. Strintzis, "3-D model search and retrieval from range images using salient features," *IEEE Trans. Multimedia*, vol. 12, no. 7, pp. 692–704, Nov. 2010.

[19] G.-H. Liu and J.-Y. Yang, "Exploiting color volume and color difference for salient region detection," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 6–16, Jan. 2019.

[20] Y. Wen, Y. Li, X. Zhang, W. Shi, L. Wang, and J. Chen, "A weighted full-reference image quality assessment based on visual saliency," *J. Vis. Commun. Image Represent.*, vol. 43, pp. 119–126, Feb. 2017.

[21] R. M. Palenichka, R. Missaoui, and M. B. Zaremba, *Extraction of Salient Features for Image Retrieval Using Multi-Scale Image Relevance Function* (Lecture Notes in Computer Science). Berlin, Germany: Springer, 2004, pp. 428–437.

[22] J. Adu, S. Xie, and J. Gan, "Image fusion based on visual salient features and the cross-contrast," *J. Vis. Commun. Image Represent.*, vol. 40, pp. 218–224, Oct. 2016.

[23] Y. Hold-Geoffroy, K. Sunkavalli, S. Hadap, E. Gambaretto, and J.-F. Lalonde, "Deep outdoor illumination estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2373–2382.

[24] S. Song and T. Funkhouser, "Neural illumination: Lighting prediction for indoor environments," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6911–6919.

[25] M.-A. Gardner, K. Sunkavalli, E. Yumer, X. Shen, E. Gambaretto, C. Gagné, and J.-F. Lalonde, "Learning to predict indoor illumination from a single image," *ACM Trans. Graph.*, vol. 36, no. 6, pp. 1–14, Nov. 2017.

[26] P. Wang and C.-Y. Chan, "Formulation of deep reinforcement learning architecture toward autonomous driving for on-ramp merge," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1–6.

[27] A. El Sallab, M. Abdou, E. Perot, and S. Yogamani, "End-to-end deep reinforcement learning for lane keeping assist," 2016, *arXiv:1612.04340*. [Online]. Available: http://arxiv.org/abs/1612.04340

[28] Y. Chen, C. Dong, P. Palanisamy, P. Mudalige, K. Muelling, and J. M. Dolan, "Attention-based hierarchical deep reinforcement learning for lane change behaviors in autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 1326–1334.

[29] X. Liang, T. Wang, L. Yang, and E. Xing, "CIRL: Controllable imitative reinforcement learning for vision-based self-driving," in *Computer Vision—ECCV*. Springer, 2018, pp. 604–620.

[30] E. Mocanu, D. C. Mocanu, P. H. Nguyen, A. Liotta, M. E. Webber, M. Gibescu, and J. G. Slootweg, "On-line building energy optimization using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 10, no. 4, pp. 3698–3708, Jul. 2019.

[31] M. Mohammadi, A. Al-Fuqaha, M. Guizani, and J.-S. Oh, "Semisupervised deep reinforcement learning in support of IoT and smart city services," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 624–635, Apr. 2018.

[32] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge computing for IoT-based energy management in smart cities," *IEEE Netw.*, vol. 33, no. 2, pp. 111–117, Mar. 2019.

[33] D. An, Q. Yang, W. Liu, and Y. Zhang, "Defending against data integrity attacks in smart grid: A deep reinforcement learning-based approach," *IEEE Access*, vol. 7, pp. 110835–110845, 2019.

[34] J. Duan, H. Li, X. Zhang, R. Diao, B. Zhang, D. Shi, X. Lu, Z. Wang, and S. Wang, "A deep reinforcement learning based approach for optimal active power dispatch," in *Proc. IEEE Sustain. Power Energy Conf. (iSPEC)*, Nov. 2019, pp. 263–267.

[35] S. Gu, E. Holly, T. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 3389–3396.

[36] F. Li, Q. Jiang, S. Zhang, M. Wei, and R. Song, "Robot skill acquisition in assembly process using deep reinforcement learning," *Neurocomputing*, vol. 345, pp. 92–102, Jun. 2019.

[37] T. Inoue, G. De Magistris, A. Munawar, T. Yokoya, and R. Tachibana, "Deep reinforcement learning for high precision assembly tasks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 819–825.

[38] B.-A. Han and J.-J. Yang, "Research on adaptive job shop scheduling problems based on dueling double DQN," *IEEE Access*, vol. 8, pp. 186474–186495, 2020.

[39] C.-C. Lin, D.-J. Deng, Y.-L. Chih, and H.-T. Chiu, "Smart manufacturing scheduling with edge computing using multiclass deep q network," *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 4276–4284, Jul. 2019.

[40] B. Waschneck, A. Reichstaller, L. Belzner, T. Altenmuller, T. Bauernhansl, A. Knapp, and A. Kyek, "Deep reinforcement learning for semiconductor production scheduling," in *Proc. 29th Annu. SEMI Adv. Semiconductor Manuf. Conf. (ASMC)*, Apr. 2018, pp. 301–306.

[41] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *J. Manuf. Syst.*, vol. 48, pp. 144–156, Jul. 2018.

[42] S. Heo and J. H. Lee, "Fault detection and classification using artificial neural networks," *IFAC-PapersOnLine*, vol. 51, no. 18, pp. 470–475, 2018.

[43] S. Zhang, S. Zhang, B. Wang, and T. G. Habetler, "Deep learning algorithms for bearing fault diagnostics—A review," in *Proc. IEEE 12th Int. Symp. Diag. Electr. Mach., Power Electron. Drives (SDEMPED)*, Aug. 2019, pp. 257–263.

[44] H. Yan, J. Wan, C. Zhang, S. Tang, Q. Hua, and Z. Wang, "Industrial big data analytics for prediction of remaining useful life based on deep learning," *IEEE Access*, vol. 6, pp. 17190–17197, 2018, doi: 10.1109/access.2018.2809681.

[45] N. Dimitriou, L. Leontaris, T. Vafeiadis, D. Ioannidis, T. Wotherspoon, G. Tinker, and D. Tzovaras, "Fault diagnosis in microelectronics attachment via deep learning analysis of 3-D laser scans," *IEEE Trans. Ind. Electron.*, vol. 67, no. 7, pp. 5748–5757, Jul. 2020.

[46] N. Dimitriou, L. Leontaris, T. Vafeiadis, D. Ioannidis, T. Wotherspoon, G. Tinker, and D. Tzovaras, "A deep learning framework for simulation and defect prediction applied in microelectronics," *Simul. Model. Pract. Theory*, vol. 100, Apr. 2020, Art. no. 102063.

[47] D. Zhang, "Review on the research and practice of deep learning and reinforcement learning in smart grids," *CSEE J. Power Energy Syst.*, vol. 4, no. 3, pp. 362–370, Sep. 2018.

[48] L. Lei, Y. Tan, K. Zheng, S. Liu, K. Zhang, and X. Shen, "Deep reinforcement learning for autonomous Internet of Things: Model, applications and challenges," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 1722–1760, 2020.

[49] H. Zhu, Y. Cao, X. Wei, W. Wang, T. Jiang, and S. Jin, "Caching transient data for Internet of Things: A deep reinforcement learning approach," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2074–2083, Apr. 2019.

[50] V. Talpaert, I. Sobh, B. Kiran, P. Mannion, S. Yogamani, A. El-Sallab, and P. Perez, "Exploring applications of deep reinforcement learning for real-world autonomous driving systems," in *Proc. 14th Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.* Setúbal, Portugal: SciTePress, 2019, pp. 564–572.

[51] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1238–1274, Aug. 2013.

[52] H. Nguyen and H. La, "Review of deep reinforcement learning for robot manipulation," in *Proc. 3rd IEEE Int. Conf. Robotic Comput. (IRC)*, Feb. 2019.

[53] S. Uddin, H. Shareef, A. Mohamed, M. A. Hannan, and K. Mohamed, "LEDs as energy efficient lighting systems: A detail review," in *Proc. IEEE Student Conf. Res. Develop.*, Dec. 2011.

[54] (Mar. 17, 2021). *Introducing Dali*. [Online]. Available: https://www.dali-alliance.org/dali/

[55] (Mar. 17, 2021). *International Electrotechnical Commission*. [Online]. Available: https://webstore.iec.ch/publication/6962

[56] A. V. Kudryashov, E. S. Galishheva, and A. S. Kalinina, "Lighting control using DALI interface," in *Proc. Int. Conf. Ind. Eng., Appl. Manuf. (ICIEAM)*, May 2018, pp. 1–5.

[57] A. Pandharipande, M. Zhao, E. Frimout, and P. Thijssen, "IoT lighting: Towards a connected building eco-system," in *Proc. IEEE 4th World Forum Internet Things (WF-IoT)*, Feb. 2018, pp. 664–669.

[58] *Simple Dali Driver Using Arduino*, Github, San Francisco, CA, USA, 2017.

[59] X. Chen, Q. Zhang, M. Lin, G. Yang, and C. He, "No-reference color image quality assessment: From entropy to perceptual quality," *EURASIP J. Image Video Process.*, vol. 2019, no. 1, p. 77, Sep. 2019.

[60] J. F. S. Gomes, F. R. Leta, P. B. Costa, and F. O. de Baldner, "Important parameters for image color analysis: An overview," in *Augmented Vision and Reality*. Berlin, Germany: Springer, 2014, pp. 81–96.

[61] K. León, D. Mery, F. Pedreschi, and J. León, "Color measurement in L∗a∗b∗ units from RGB digital images," *Food Res. Int.*, vol. 39, no. 10, pp. 1084–1091, Dec. 2006.

[62] F. Garcia and E. Rachelson, "Markov decision processes," in *Markov Decision Processes in Artificial Intelligence*. Hoboken, NJ, USA: Wiley, Mar. 2013, pp. 1–38.

[63] Z. Wang and Q. Li, "Information content weighting for perceptual image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 5, pp. 1185–1198, May 2011.

[64] H.-I. Lin and P.-Y. Lin, "An image quality assessment method for surface defect inspection," in *Proc. IEEE Int. Conf. Artif. Intell. Test. (AITest)*, Aug. 2020, pp. 1–6.

LAMPROS LEONTARIS (Member, IEEE) received the Diploma degree in electrical and computer engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2016. Since March 2017, he has been a Research Assistant with the Information Technologies Institute, Centre for Research and Technology Hellas, Thessaloniki. His research interests include computational intelligence methods, machine learning, and fuzzy systems.

NIKOLAOS DIMITRIOU (Member, IEEE) received the Diploma and Ph.D. degrees in electrical and computer engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2007 and 2014, respectively. Since July 2014, he has been a Postdoctoral Research Associate with the Informatics and Telematics Institute, Centre for Research and Technology Hellas, Thessaloniki. His research interests include machine vision and deep learning with a focus on industrial applications.

DIMOSTHENIS IOANNIDIS received the Diploma degree in electrical and computing engineering and the master's degree in advanced communication systems and engineering from the Aristotle University of Thessaloniki (AUTh), Thessaloniki, Greece, in 2000 and 2005, respectively. He is currently a Senior Researcher Grade C' with the Informatics and Telematics Institute, Centre for Research and Technology Hellas, Thessaloniki. His main research interests include computer vision, stereoscopic image processing and signal analysis, linguistics algorithms, Web services, semantics, visual analytics, smart grids and energy efficiency, the IoT platforms development, and security ecosystems, including blockchain and research in ethics and biometrics.

KONSTANTINOS VOTIS received the M.Sc. and Ph.D. degrees in computer science and service oriented architectures from the Computer Engineering and Informatics Department, University of Patras, Greece, and the M.B.A. degree from the Department of Business School, University of Patras. He is currently a Senior Researcher Grade B' with the Informatics and Telematics Institute, Centre for Research and Technology Hellas, Thessaloniki, Greece, and the Director of the Visual Analytics Laboratory. His research interests include human–computer interaction (HCI), information visualisation and management of big data, knowledge engineering and decision support systems, the Internet of Things, cybersecurity, and pervasive computing with major application areas, such as mHealth, eHealth, and personalized healthcare.

DIMITRIOS TZOVARAS (Senior Member, IEEE) received the Diploma and Ph.D. degrees in electrical and computer engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1992 and 1997, respectively. He is currently a Senior Researcher Grade A' and the President of the Board with the Centre for Research and Technology Hellas, Thessaloniki. His main research interests include visual analytics, three-dimensional object recognition, search and retrieval, behavioral biometrics, assistive technologies, information and knowledge management, computer graphics, and virtual reality.

ELPINIKI PAPAGEORGIOU (Senior Member, IEEE) received the M.Sc. degree in medical physics and the Ph.D. degree in computer science from the University of Patras, in 2000 and 2004, respectively. She is currently an Associate Professor with the Energy Systems Department, University of Thessaly at Geopolis, Larissa, Greece. She specializes in developing and applying artificial intelligent models and algorithms to decision support problems for modeling, prediction, strategic decisions, scenario analysis and data mining, and solving important emerging problems arising in engineering, energy, business, medicine, agriculture, and environment. She has more than 250 publications in journal articles, conference papers, and book chapters (100 of them are in journals with high impact factors). She has more than 4800 citations from independent researchers (H-index=37 in Scopus and 47 in GoogleScholar). She is also within the 2% among 100,000 overall top-cited most influential scientists, regarding her composite citation index across all scientists and scientific disciplines, according to the published catalogues by PLoS Biology 2020 & Mendeley Data 2020. Her main research interests include development of novel algorithms and fuzzy models for intelligent decision support systems focused on fuzzy cognitive maps.