# Real-Time Dense Stereo Embedded in A UAV for Road Inspection

Rui Fan, Jianhao Jiao, Jie Pan, Huaiyang Huang, Shaojie Shen, Ming Liu
HKUST Robotics Institute
rui.fan@ieee.org

## Abstract

*The condition assessment of road surfaces is essential to ensure their serviceability while still providing maximum road traffic safety. This paper presents a robust stereo vision system embedded in an unmanned aerial vehicle (UAV). The perspective view of the target image is first transformed into the reference view, and this not only improves the disparity accuracy, but also reduces the algorithm's computational complexity. The cost volumes generated from stereo matching are then filtered using a bilateral filter. The latter has been proved to be a feasible solution for the functional minimisation problem in a fully connected Markov random field model. Finally, the disparity maps are transformed by minimising an energy function with respect to the roll angle and disparity projection model. This makes the damaged road areas more distinguishable from the road surface. The proposed system is implemented on an NVIDIA Jetson TX2 GPU with CUDA for real-time purposes. It is demonstrated through experiments that the damaged road areas can be easily distinguished from the transformed disparity maps.*

## 1. Introduction

The frequent detection of different types of road damage, *e.g.*, cracks and potholes, is a critical task in road maintenance [21]. Road condition assessment reports allow governments to appraise long-term investment schemes and allocate limited resources for road maintenance [5]. However, manual visual inspection is still the main form of road condition assessment [15]. This process is, however, not only tedious, time-consuming and costly, but also dangerous for the personnel [16]. Furthermore, the detection results are always subjective and qualitative because decisions entirely depend on the experience of the personnel [17]. Therefore, there is an ever-increasing need to develop automated road inspection systems that can recognise and localise road damage both efficiently and objectively [21].

Over the past decades, various technologies, such as vibration sensing, active or passive sensing, have been used to acquire road data and help technicians in assessing the road condition [18]. For example, Fox *et al*. [9] developed a crowd-sourcing system to detect road damage by analysing accelerometer data obtained from multiple vehicles. Although vibration sensors are cost-effective and only require a small amount of storage space, the shape of a damaged road area cannot be explicitly inferred from the vibration data [15]. Furthermore, Tsai *et al*. [28] mounted two laser scanners on a digital inspection vehicle (DIV) to collect 3D road data for pothole detection. However, such vehicles are not widely used, because of their high equipment and long-term maintenance costs [5].

The most commonly used passive sensors for road condition assessment include Microsoft Kinect and other types of digital cameras [30]. In [14], Jahanshahi *et al*. utilised a Kinect to acquire depth maps, from which the damaged road areas were extracted using image segmentation algorithms. However, Kinect sensors were initially designed for indoor use, and they do not perform well when exposed to direct sunlight, causing depth values to be recorded as zero [3]. Therefore, it is more effective to detect road damages using digital cameras, as they are cost-effective and capable of working in outdoor environments [5].

With recent advances in airborne technology, unmanned aerial vehicles (UAVs) equipped with digital cameras provide new opportunities for road inspection [25]. For example, Feng *et al*. [8] mounted a camera on a UAV to capture road images. The latter was then analysed to illustrate conditions such as traffic congestion, road accidents, among others. Furthermore, Zhang [34] designed a robust photogrammetric mapping system for UAVs, which can recognise different road defects, such as ruts and potholes, from the captured RGB images. Although the aforementioned 2D computer vision methods can recognise damaged road areas with low computational complexity, the achieved level of accuracy is still far from satisfactory [14, 16]. Additionally, the structure of a detected road damage is not obvious from only a single video frame, and the depth/disparity information is more effective than RGB information in terms of detecting severe road damages, *e.g.*, potholes [21]. Therefore, it becomes increasingly important to use digital cameras for 3D road data acquisition.

To reconstruct 3D road scenery using digital cameras, multiple camera views are required [11]. Images from different viewpoints can be captured using either a single movable camera or an array of synchronised cameras [5]. In [35], Zhang and Elaksher reconstructed the 3D road scenery using structure from motion (SfM), where the keypoints in each frame were extracted using scale-invariant feature transform (SIFT) [19], and an energy function with respect to all camera poses was optimised for accurate 3D road scenery reconstruction. However, SfM can only acquire sparse point clouds, which are usually infeasible for road damage detection [14]. In this regard, many researchers have resorted to using stereo vision technology to acquire dense point clouds for road damage detection. In [5], Fan *et al*. developed an accurate dense stereo vision algorithm for road surface 3D reconstruction, and an accuracy of approximately $\pm$ 3 mm was achieved. However, the search range propagation strategy in their algorithm makes it difficult to fully exploit the parallel computing architecture of the graphics cards [5]. Therefore, the motivation of this paper is to explore a highly efficient dense stereo vision algorithm, which can be embedded in UAVs for real-time road inspection.

The remainder of this paper is organised as follows. Section 2 discusses the related work on stereo vision. Section 3 presents the proposed embedded stereo vision system. The experimental results for performance evaluation are provided in Section 4. Finally, Section 5 summarises the paper and provides recommendations for future work.

## 2. Related Work

The two key aspects of computer stereo vision are speed and accuracy [27]. A lot of research has been carried out over the past decades to improve either the disparity accuracy or the algorithm's computational complexity [5]. The state-of-the-art stereo vision algorithms can be classified as convolutional neural network (CNN)-based [2,20,32,33,36] and traditional [1,5,12,13,23,26]. The former generally formulates disparity estimation as a binary classification problem and learns the probability distribution over all disparity values [20]. For example, PSMNet [2] generates the cost volumes by learning region-level features with different scales of receptive fields. Although these approaches have achieved some highly accurate disparity maps, they usually require a large amount of labelled training data to learn from. Therefore, it is impossible for them to work on the datasets without providing the disparity ground truth [36]. Moreover, predicting disparities with CNNs is still a computationally intensive task, which usually takes seconds or even minutes to execute on state-of-the-art graphics cards [27]. Therefore, the existing CNN-based stereo vision algorithms are not suitable for real-time applications.

The traditional stereo vision algorithms can be classified as local, global and semi-global [5]. The local algorithms typically select a series of blocks from the target image and match them with a constant block selected from the reference image [5]. The disparities are then determined by finding the shifting distances corresponding to either the highest correlation or the lowest cost [27]. This optimisation technique is also known as winner-take-all (WTA).

Unlike the local algorithms, the global algorithms generally translate stereo matching into an energy minimisation problem, which can later be addressed using sophisticated optimisation techniques, *e.g*., belief propagation (BP) [13] and graph cuts (GC) [1]. These techniques are commonly developed based on the Markov random field (MRF) [26]. Semi-global matching (SGM) [12] approximates the MRF inference by performing cost aggregation along all directions in the image, and this greatly improves the accuracy and efficiency of stereo matching. However, finding the optimum smoothness values is a challenging task, due to the occlusion problem [23]. Over-penalising the smoothness term can reduce ambiguities around the discontinuous areas, but on the other hand, can cause incorrect matches for the continuous areas [5]. Furthermore, the computational complexities of the aforementioned optimisation techniques are significantly intensive, making these algorithms difficult to perform in real time [27].

In [5], Fan *et al*. proposed a novel perspective transformation method, which improves both the disparity accuracy and the computational complexity of the algorithm. Furthermore, Mozerov and Weijer [23] proved that bilateral filtering is a feasible solution for the energy minimisation problem in a fully connected MRF model. The costs can be adaptively aggregated by performing bilateral filtering on the initial cost volumes [5]. Therefore, the proposed stereo vision system is developed based on the work in [5] and [23]. Finally, the estimated disparity maps are transformed by minimising an energy function with respect to the roll angle and disparity projection model. This makes the damaged road areas become highly distinguishable from the road surface.

## 3. System Description

The workflow of the proposed stereo vision system is depicted in Figure 1, where the system consists of three main components: a) perspective transformation; b) dense road stereo; and c) disparity transformation. The following subsections describe each component in turn.

### 3.1. Perspective Transformation

In this paper, the road surface is treated as a ground plane:

$$\mathbf{n}^{\top}\mathbf{p}^{\mathrm{W}} + \beta = 0, \tag{1}$$

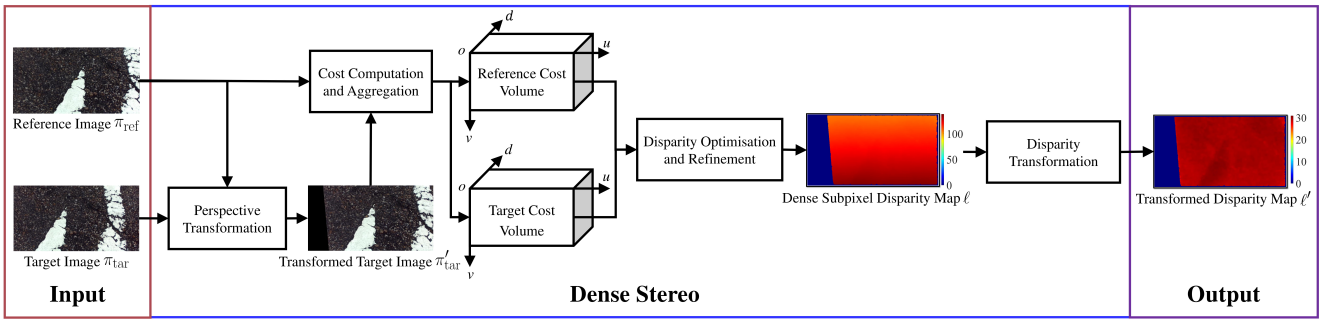


Figure 1. Workflow of the proposed dense stereo system.

where $\mathbf{p}^{\text{W}} = [x^{\text{W}}, y^{\text{W}}, z^{\text{W}}]^\top$ is a 3D point on the road surface in the world coordinate system (WCS), and $\mathbf{n} = [n_x, n_y, n_z]^\top$ is the normal vector of the ground plane. The projections of $\mathbf{p}^{\text{W}}$ on the reference and target images, *i.e.*, $\pi_{\text{ref}}$ and $\pi_{\text{tar}}$, are $\mathbf{p}^{\text{I}}_{\text{ref}} = [u_{\text{ref}}, v_{\text{ref}}]^\top$ and $\mathbf{p}^{\text{I}}_{\text{tar}} = [u_{\text{tar}}, v_{\text{tar}}]^\top$, respectively. It should be noted that the left and right images are respectively referred to as the reference and target images in this paper. $\mathbf{p}^{\text{I}}_{\text{ref}}$ can be transformed to $\mathbf{p}^{\text{I}}_{\text{tar}}$ using a homography matrix $\mathbf{H}$ as follows [5]:

$$\begin{bmatrix} \mathbf{p}^{\text{I}}_{\text{tar}} \\ 1 \end{bmatrix} = \mathbf{H} \begin{bmatrix} \mathbf{p}^{\text{I}}_{\text{ref}} \\ 1 \end{bmatrix}, \tag{2}$$

where

$$\mathbf{H} = \mathbf{K}_{\text{tar}} \left( \mathbf{R}_1 - \frac{\mathbf{t}\mathbf{n}^\top \mathbf{R}_0{}^{-1}}{\beta} \right) \mathbf{K}_{\text{ref}}{}^{-1}, \tag{3}$$

$\mathbf{t}$ is a translation vector, $\mathbf{R}_0$ represents the rotation from the WCS to the reference camera coordinate system (RCCS), $\mathbf{R}_1$ denotes the rotation from the RCCS to the target camera coordinate system (TCCS), and $\mathbf{K}_{\text{ref}}$ and $\mathbf{K}_{\text{tar}}$ are the intrinsic matrices of the reference and target cameras, respectively. $\mathbf{H}$ can be estimated using at least four pairs of matched correspondence points $\mathbf{p}^{\text{I}}_{\text{ref}}$ and $\mathbf{p}^{\text{I}}_{\text{tar}}$ [11]. In order to simplify the estimation of $\mathbf{H}$, the authors of [5] made several hypotheses regarding $\mathbf{R}_0$, $\mathbf{R}_1$, $\mathbf{K}_{\text{ref}}$, $\mathbf{K}_{\text{tar}}$, $\mathbf{t}$ and $\mathbf{n}$. (2) can be rewritten as follows:

$$u_{\text{tar}} = u_{\text{ref}} + \frac{t_c n_x}{\beta}(f \sin\theta - v_o \cos\theta) + v\frac{t_c n_x}{\beta}\cos\theta, \tag{4}$$

where $f$ is the focus length of each camera, $t_c$ is the baseline, $\theta$ is the pitch angle, and $[u_o, v_o]^\top$ is the principal point. $v = v_{\text{ref}} = v_{\text{tar}}$. (4) implies that a perspective distortion always exists for the ground plane in two images when $\theta$ is not equal to $\pi/2$, and this further affects the stereo matching accuracy. Therefore, the perspective transformation aims to make the ground plane in the transformed target image similar to that in the reference image [5]. This can be straightforwardly realised by shifting each point on row $v$ in the target image $\Delta u - \delta_p$ pixels to the right, where $\Delta u = u_{\text{ref}} - u_{\text{tar}}$, and $\delta_p$ is a constant used to guarantee that all the disparities are non-negative. The values of $t_c$, $n_x$, $f$, $\beta$, $v_o$ and $\theta$ can be estimated from a set of reliable correspondence pairs $\mathbf{Q}_{\text{ref}} = [\mathbf{p}^{\text{I}}_{\text{ref}_0}, \mathbf{p}^{\text{I}}_{\text{ref}_1}, \ldots, \mathbf{p}^{\text{I}}_{\text{ref}_n}]^\top$ and $\mathbf{Q}_{\text{tar}} = [\mathbf{p}^{\text{I}}_{\text{tar}_0}, \mathbf{p}^{\text{I}}_{\text{tar}_1}, \ldots, \mathbf{p}^{\text{I}}_{\text{tar}_n}]^\top$. The transformed target image is shown in Figure 1 as $\pi'_{\text{tar}}$.

## 3.2. Dense Road Stereo

### 3.2.1 Cost Computation and Aggregation

According to [23], finding the best disparities is equivalent to maximising the joint probability in (5):

$$P(\mathbf{p}_{ij}, q) = \prod_{\mathbf{p}_{ij} \in \mathscr{P}} \Phi(\mathbf{p}_{ij}, q_{\mathbf{p}_{ij}}) \prod_{\mathbf{n}_{\mathbf{p}_{ij}} \in \mathscr{N}_{\mathbf{p}_{ij}}} \Psi(\mathbf{p}_{ij}, \mathbf{n}_{\mathbf{p}_{ij}}), \tag{5}$$

where $\mathbf{p}_{ij}$ denotes a node at the position of $(i, j)$ in the graph $\mathscr{P}$, $q_{\mathbf{p}_{ij}}$ represents the intensity differences corresponding to different disparities $d$, $\mathscr{N}_{\mathbf{p}_{ij}} = \{\mathbf{n}_{\mathbf{p}_{ij_1}}, \mathbf{n}_{\mathbf{p}_{ij_2}}, \mathbf{n}_{\mathbf{p}_{ij_3}}, \cdots, \mathbf{n}_{\mathbf{p}_{ij_k}} | \mathbf{n}_{\mathbf{p}_{ij}} \in \mathscr{P}\}$ represents the neighbourhood system of $\mathbf{p}_{ij}$, $\Phi(\cdot)$ expresses the compatibility between each possible disparity $d$ and the corresponding intensity difference, and $\Psi(\cdot)$ expresses the compatibility between $\mathbf{p}_{ij}$ and its neighbourhood system $\mathscr{N}_{\mathbf{p}_{ij}}$. It is noteworthy that $\mathbf{p}_{uv}$ refers to $\mathbf{p}^{\text{I}}_{\text{ref}} = [u_{\text{ref}}, v_{\text{ref}}]^\top$ and $\mathscr{P}$ refers to the reference image. In practice, maximising the joint probability in (5) is commonly formulated as an energy minimisation problem as follows [7]:

$$E_d(\mathbf{p}_{ij}, d) = \sum_{\mathbf{p}_{ij} \in \mathscr{P}} D(\mathbf{p}_{ij}, d) + \sum_{\mathbf{n}_{\mathbf{p}_{ij}} \in \mathscr{N}_{\mathbf{p}_{ij}}} V(\mathbf{p}_{ij}, \mathbf{n}_{\mathbf{p}_{ij}}, d), \tag{6}$$

where $D(\cdot)$ computes the matching cost of $\mathbf{p}_{ij}$, and $V(\cdot)$ determines the aggregation strategy. For disparity estimation algorithms based on the MRF, formulating $V(\cdot)$ in an adaptive way is crucial and necessary, because the intensity of a pixel in a discontinuous area usually differs greatly from those of its neighbours [23]. Since bilateral filtering is a feasible solution for the energy minimisation problem in a fully connected MRF model [23], $D(\cdot)$ and $V(\cdot)$ can be rewritten as follows:

$$D(\mathbf{p}_{ij}, d) = c(\mathbf{p}_{ij}, d), \tag{7}$$

where

$$c(\mathbf{p},d)=\frac{(\sigma_{\text{ref}}\sigma_{\text{tar}}+\mu_{\text{ref}}\mu_{\text{tar}})}{\sigma_{\text{ref}}\sigma_{\text{tar}}} -\frac{1}{n\sigma_{\text{ref}}\sigma_{\text{tar}}}\left(\sum_{\mathbf{q}\in\mathcal{N}_{\mathbf{p}}^{+}} i_{\text{ref}}(\mathbf{q})i_{\text{tar}}(\mathbf{q}-[d,0]^{\top})\right) \tag{8}$$

is the cost function; $i_{\text{ref}}(\mathbf{p})$ and $i_{\text{tar}}(\mathbf{p})$ represent the pixel intensities at $\mathbf{p}$ in the reference and target images, respectively; $\mu_{\text{ref}}$ and $\mu_{\text{tar}}$ represent the means of the pixel intensities within the reference and target blocks, respectively; and $\sigma_{\text{ref}}$ and $\sigma_{\text{tar}}$ denote the standard deviations of the reference and target blocks, respectively. $\mathcal{N}_{\mathbf{p}}^{+}=\{\mathbf{p}\}\cup\mathcal{N}_{\mathbf{p}}$.

$$V(\mathbf{p}_{ij},\mathbf{n}_{\mathbf{p}_{ij}},d)=\sum_{\mathbf{n}_{\mathbf{p}_{ij}}\in\mathcal{N}_{\mathbf{p}_{ij}}}\omega(\mathbf{p}_{ij},\mathbf{n}_{\mathbf{p}_{ij}})c(\mathbf{n}_{\mathbf{p}_{ij}},d), \tag{9}$$

where

$$\omega(\mathbf{p},\mathbf{n}_{\mathbf{p}})=\exp\left\{-\frac{\|\mathbf{p}-\mathbf{n}_{\mathbf{p}}\|_2^2}{{\sigma_0}^2}-\frac{(i_{\text{ref}}(\mathbf{p})-i_{\text{ref}}(\mathbf{n}_{\mathbf{p}}))^2}{{\sigma_1}^2}\right\} \tag{10}$$

is controlled by two parameters $\sigma_0$ and $\sigma_1$, with $\sigma_0$ based on spatial distance and $\sigma_1$ based on colour similarity. The cost $c$ of each neighbour $\mathbf{n}_{\mathbf{p}}$ can therefore be adaptively aggregated to $\mathbf{p}$. Finally, $E_d(\mathbf{p},d)$ is normalised by rewriting (6) as follows:

$$E_d(\mathbf{p},d)=\frac{\sum_{\mathbf{q}\in\mathcal{N}_{\mathbf{p}}^{+}}\omega(\mathbf{p},\mathbf{q})D(\mathbf{q},d)}{\sum_{\mathbf{q}\in\mathcal{N}_{\mathbf{p}}^{+}}\omega(\mathbf{p},\mathbf{q})}, \tag{11}$$

The computed matching costs are stored in two cost volumes, as shown in Figure 1.

#### 3.2.2 Disparity Optimisation and Refinement

By applying WTA optimisation on the reference and target cost volumes, the best disparities can be estimated. Since the perspective view of the target image has been transformed in Section 3.1, the estimated disparities on row $v$ should be added $\Delta u-\delta_p$ to obtain the disparity map between the original reference and target images. The occluded areas in the reference disparity map are then removed by finding the pixels $\mathbf{p}$ satisfying the following condition [4]:

$$\left\|\ell_{\text{ref}}(\mathbf{p})-\ell_{\text{tar}}(\mathbf{p}-[\ell_{\text{ref}}(\mathbf{p}),0]^{\top})\right\|_2^2>\delta_r, \tag{12}$$

where $\ell_{\text{ref}}$ and $\ell_{\text{tar}}$ represent the reference and target disparity maps, respectively. $\delta_r=1$ is the threshold for occlusion removal. Finally, a subpixel enhancement is performed to increase the resolution of the estimated disparity values [5]:

$$\ell(\mathbf{p})=\ell_{\text{ref}}(\mathbf{p})+\frac{c(\mathbf{p},d-1)-c(\mathbf{p},d+1)}{2c(\mathbf{p},d-1)+2c(\mathbf{p},d+1)-4c(\mathbf{p},d)}, \tag{13}$$

where $\ell$, illustrated in Figure 1, represents the final disparity map in the reference perspective view.

### 3.3. Disparity Transformation

The proposed system focuses entirely on the road surface whose disparity values decrease gradually from the bottom of the disparity map to its top, as shown in Figure 1. For a stereo rig whose baseline is perfectly parallel to the road surface, the roll angle $\psi$ equals zero and the disparities on each row have similar values, which can also be proved by (4). Therefore, the projection of the road disparities on a v-disparity image can be represented by a linear model: $f(v)=\alpha_0+\alpha_1 v$. A column vector $\boldsymbol{\alpha}=[\alpha_0,\alpha_1]^{\top}$ storing the coefficients of the disparity projection model can be estimated as follows:

$$\boldsymbol{\alpha}=\arg\min_{\boldsymbol{\alpha}} E_t, \tag{14}$$

where

$$E_t=\|\mathbf{d}-\mathbf{V}\boldsymbol{\alpha}\|_2^2, \tag{15}$$

$\mathbf{d}=[\ell(\mathbf{p}_0),\ \ell(\mathbf{p}_1),\ \cdots,\ \ell(\mathbf{p}_n)]^{\top}$ stores the disparity values, $\mathbf{v}=[v_0,\ v_1,\ \cdots,\ v_n]^{\top}$ stores the vertical disparity coordinates, $\mathbf{1}_k$ represents a $k\times 1$ vector of ones, and $\mathbf{V}=[\mathbf{1}_{n+1}\ \mathbf{v}]$. Applying (15) to (14) results in the following expression:

$$\boldsymbol{\alpha}=(\mathbf{V}^{\top}\mathbf{V})^{-1}\mathbf{V}^{\top}\mathbf{d}. \tag{16}$$

The minimum energy $E_{t\text{min}}$ can be obtained by applying (16) to (15):

$$E_{t\text{min}}=\mathbf{d}^{\top}\mathbf{d}-\mathbf{d}^{\top}\mathbf{V}(\mathbf{V}^{\top}\mathbf{V})^{-1}\mathbf{V}^{\top}\mathbf{d}. \tag{17}$$

However, in practice, the stereo rig baseline is not always perfectly parallel to the road surface, and this introduces a non-zero roll angle $\psi$ into the imaging process. The disparity values will change gradually in the horizontal direction, and this makes the approach of representing the road disparity projection using a linear model problematic. Additionally, the minimum energy $E_{t\text{min}}$ becomes higher, due to the disparity dispersion in the horizontal direction. Hence, the proposed disparity transformation first finds the angle corresponding to the minimum $E_{t\text{min}}$. The image rotation caused by $\psi$ is then eliminated, and $\boldsymbol{\alpha}$ is subsequently estimated.

To rotate the disparity map around a given angle $\psi$, each set of original coordinates $[u,v]^{\top}$ is transformed to a set of new coordinates $[x(\psi),y(\psi)]^{\top}$ using the following equations [6]:

$$x(\psi)=u\cos\psi+v\sin\psi, \tag{18}$$

$$y(\psi)=v\cos\psi-u\sin\psi. \tag{19}$$

The energy function in (15) can, therefore, be rewritten as follows:

$$E_t(\psi)=\|\mathbf{d}-\mathbf{Y}(\psi)\boldsymbol{\alpha}\|_2^2, \tag{20}$$

where $\mathbf{y} = [y_0(\psi),\ y_1(\psi),\ \cdots,\ y_n(\psi)]^\top$ and $\mathbf{Y}(\psi) = [\mathbf{1}_{n+1}\ \mathbf{y}(\psi)]$. (21) is obtained by applying (20) to (14):

$$\boldsymbol{\alpha}(\psi) = \mathbf{J}(\psi)\mathbf{d}, \tag{21}$$

where

$$\mathbf{J}(\psi) = (\mathbf{Y}(\psi)^\top \mathbf{Y}(\psi))^{-1}\mathbf{Y}(\psi)^\top. \tag{22}$$

$E_{t\min}$ can also be obtained by applying (21) and (22) to (20):

$$E_{t\min}(\psi) = \mathbf{d}^\top\mathbf{d} - \mathbf{d}^\top\mathbf{Y}\big(\mathbf{Y}(\psi)^\top\mathbf{Y}(\psi)\big)^{-1}\mathbf{Y}(\psi)^\top\mathbf{d}. \tag{23}$$

Roll angle estimation is, therefore, equivalent to the following energy minimisation problem:

$$\psi = \arg\min_{\psi} E_{t\min}(\psi) \ \text{ s.t. } \psi \in (-\frac{\pi}{2}, \frac{\pi}{2}], \tag{24}$$

which can be formulated as an iterative optimisation problem as follows [24]:

$$\psi^{(k+1)} = \psi^{(k)} - \lambda\nabla E_{t\min}(\psi^{(k)}), \quad k \in \mathbb{N}^0, \tag{25}$$

where $\lambda$ is the learning rate. (25) is a standard form of gradient descent. The expression of $\nabla E_{t\min}$ is as follows:

$$\nabla E_{t\min}(\psi) = -2\mathbf{d}^\top\mathbf{W}(\psi)\mathbf{d}, \tag{26}$$

where

$$\mathbf{W}(\psi) = \Big(\mathbf{I} - \mathbf{Y}(\psi)\mathbf{J}(\psi)\Big)\nabla\mathbf{Y}(\psi)\mathbf{J}(\psi), \tag{27}$$

$\mathbf{I}$ is an identity matrix. If $\lambda$ is too high, (25) may overshoot the minimum. On the other hand, if $\lambda$ is set to a relatively low value, the convergence of (25) may require a lot of iterations [24]. Therefore, selecting a proper $\lambda$ is always essential for gradient descent. Instead of fixing the learning rate with a constant value, backtracking line search is utilised to produce an adaptive learning rate:

$$\lambda^{(k+1)} = \frac{\lambda^{(k)}\nabla E_{t\min}(\psi^{(k)})}{\nabla E_{t\min}(\psi^{(k)}) - \nabla E_{t\min}(\psi^{(k+1)})}, \quad k \in \mathbb{N}^0. \tag{28}$$

The selection of the initial learning rate $\lambda^{(0)}$ will be discussed in Section 4. The initial approximation $\psi^{(0)}$ is set to 0, because the roll angle in practical experiments is usually small. It should be noted that the estimated $\psi$ at time $t$ is used as the initial approximation at time $t+1$. The optimisation iterates until the absolute difference between $\psi^{(k)}$ and $\psi^{(k+1)}$ is smaller than a preset threshold $\delta_\psi$. $\boldsymbol{\alpha}$ can be obtained by substituting the estimated roll angle $\psi$ into (21). Finally, each disparity is transformed using:

$$\ell'(\mathbf{p}) = \ell(\mathbf{p}) - \alpha_0 + \alpha_1(u\sin\psi - v\cos\psi) + \delta_t, \tag{29}$$

where $\ell'$, shown in Figure 1, represents the transformed disparity map, and $\delta_t$ is a constant used to make the transformed disparity values positive.

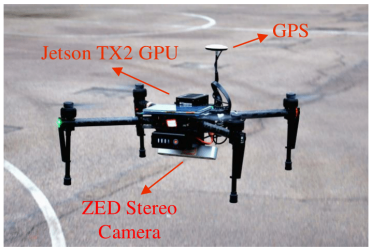


Figure 2. Experimental set-up.

## 4. Experimental Results

In this section, we evaluate the performance of the proposed stereo vision system both qualitatively and quantitatively. The following subsections detail the experimental set-up, datasets, implementation notes and the performance evaluation.

### 4.1. Experimental Set-Up

In the experiments, a ZED stereo camera[1] is mounted on a DJI Matrice 100 Drone[2] to capture stereo road images. The maximum take-off weight of the drone is 3.6 kg. The stereo camera has two ultra-sharp six-element all-glass lenses, which can cover the scene up to 20 m[1]. The captured stereo road images are processed using an NVIDIA Jetson TX2 GPU[3], which has 8 GB LPDDR4 memory and 256 CUDA cores. An illustration of the experimental set-up is shown in Figure 2.

### 4.2. Datasets

Using the above experimental set-up, three datasets including 11368 stereo image pairs are created. The resolution of the original reference and target images is $640 \times 360$. In each dataset, the UAV flight trajectory forms a closed loop, which makes it possible to evaluate the performance of the state-of-the-art visual odometry algorithms using our created datasets. The datasets and a demo video are publicly available at http://www.ruirangerfan.com.

### 4.3. Implementation Notes

In the practical implementation, the reference and target images are first sent to the global memory of the GPU from the host memory. However, a thread is more likely to fetch the data from the closest addresses that its nearby threads accessed[4]. This fact makes the use of cache in global memory impossible. Furthermore, constant memory and texture memory are read-only and cached on-chip, and this makes them more efficient than global memory for memory requesting[4]. Therefore, we store the reference and target im-

[1] https://www.stereolabs.com/
[2] https://www.dji.com/uk/matrice100
[3] https://developer.nvidia.com/embedded/buy/jetson-tx2
[4] https://docs.nvidia.com/cuda/pdf/CUDA_C_Programming_Guide.pdf

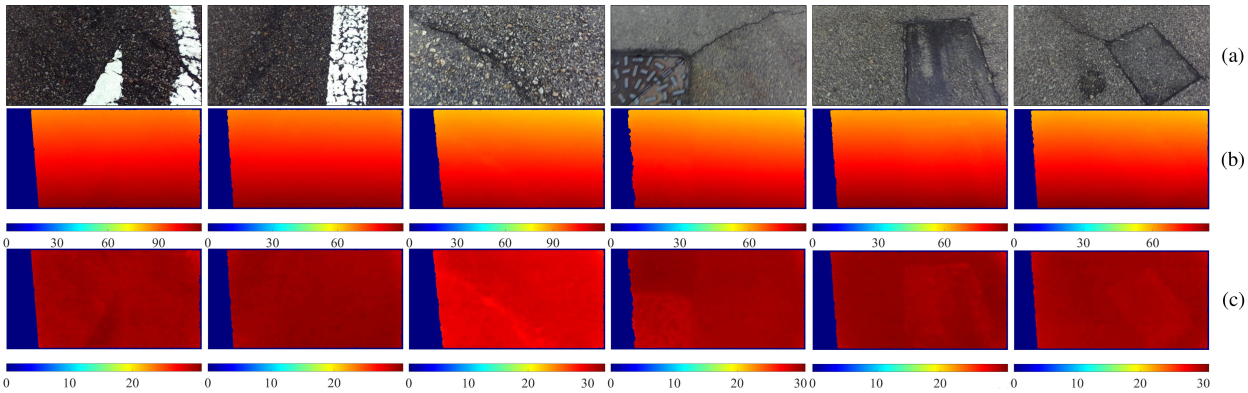


Figure 3. Experimental results; (a) reference images; (b) dense subpixel disparity maps; (c) transformed disparity maps.

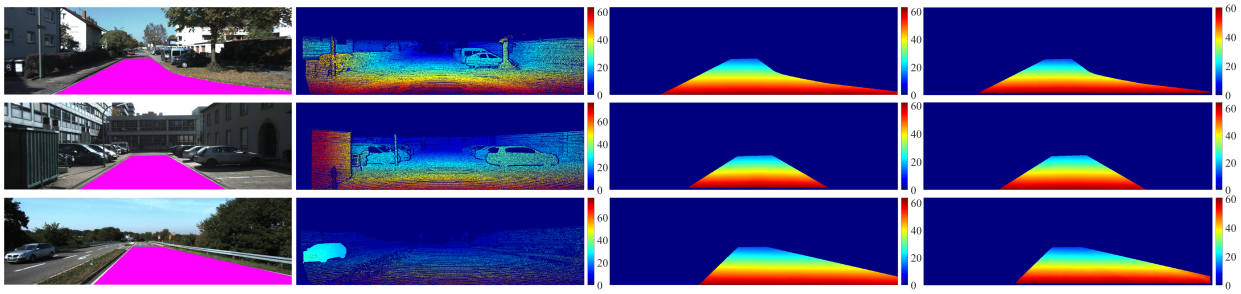


Figure 4. Examples of the KITTI stereo experimental results; (a) reference images, where the areas in magenta are the manually selected road regions; (b) ground truth disparity maps; (c) results obtained using PSMNet; (d) results obtained using the proposed algorithm.

ages in the texture memory to reduce the memory requests from the global memory. This is realised by creating two texture objects in the texture memory and binding these objects with the addresses of the reference and target images. The pixel intensities can therefore be fetched from the texture objects instead of the global memory. In addition, (10) is rewritten as follows:

$$\omega(\mathbf{p},\mathbf{n_p}) = \omega_0(\mathbf{p},\mathbf{n_p})\omega_1(\mathbf{p},\mathbf{n_p}), \tag{30}$$

where

$$\omega_0(\mathbf{p},\mathbf{n_p}) = \exp\left\{-\frac{\|\mathbf{p}-\mathbf{n_p}\|_2^2}{{\sigma_0}^2}\right\} \tag{31}$$

and

$$\omega_1(\mathbf{p},\mathbf{n_p}) = \exp\left\{-\frac{(i_{\text{ref}}(\mathbf{p})-i_{\text{ref}}(\mathbf{n_p}))^2}{{\sigma_1}^2}\right\}. \tag{32}$$

The values of $\omega_0$ and $\omega_1$ are pre-calculated and stored in the constant memory to reduce the repetitive computations of $\omega$. Moreover, the values of $\mu_{\text{ref}}$, $\mu_{\text{tar}}$, $\sigma_{\text{ref}}$ and $\sigma_{\text{tar}}$ are also pre-calculated and stored in the global memory to avoid the unnecessary computations in stereo matching.

## 4.4. Performance Evaluation

### 4.4.1 Disparity Estimation

Some experimental results are illustrated in Figure 3. $\mathcal{N}$ is a 120-connected neighbourhood system. $\sigma_0$ and $\sigma_1$ are empirically set to 1.5 and 5.5, respectively. Since the datasets we created do not contain disparity ground truth, the KITTI[5] stereo 2012 and 2015 datasets [10, 22] are utilised to quantify the accuracy of the proposed system. Some experimental results of the KITTI stereo datasets are shown in Figure 4, where the road regions are manually selected to evaluate the accuracy of the road disparities. Furthermore, we compare the proposed method with PSMNet [2] in terms of the percentage of error pixels $e_p$ and root mean squared error $e_r$. The expressions of $e_p$ and $e_r$ are as follows:

$$e_p = \frac{1}{m}\sum_{\mathbf{p}} \delta\big(|\ell(\mathbf{p})-\tilde{\ell}(\mathbf{p})|,\varepsilon_d\big)\times 100\%, \tag{33}$$

$$e_r = \sqrt{\frac{1}{m}\sum_{\mathbf{p}}\big(\ell(\mathbf{p})-\tilde{\ell}(\mathbf{p})\big)^2}, \tag{34}$$

where

$$\delta(x,\varepsilon_d) = \begin{cases} 1 & (x>\varepsilon_d) \\ 0 & (x\le\varepsilon_d) \end{cases}, \tag{35}$$

[5] http://www.cvlibs.net/datasets/kitti/

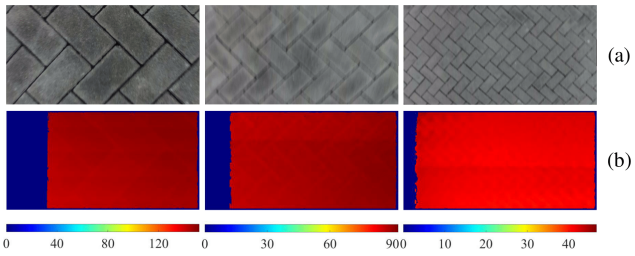


Figure 5. Disparity maps of some motion blurred images; (a) reference images; (b) disparity maps.

$m$ is the total number of disparities used for evaluation, $\varepsilon_d$ is the disparity error tolerance, and $\tilde{\ell}$ represents the ground truth disparity map. The comparison of $e_p$ and $e_r$ between these two methods is shown in Table 1, where it can be observed that the proposed method outperforms PSMNet in terms of $e_p$ and $e_r$ when $\varepsilon_d$ is set to 2, while PSMNet performs better than our method when $\varepsilon_d$ is set to 3. It should be noted that the proposed algorithm is capable of estimating disparity maps between a pair of motion blurred stereo images, as shown in Figure 5. This also demonstrates the robustness of the proposed dense stereo system.

| Method | $e_r$ | $e_p$ | |
|---|---|---|---|
| | | $\varepsilon_d = 2$ | $\varepsilon_d = 3$ |
| PSMNet | 1.039 | 1.345 | 0.016 |
| Ours | 0.409 | 0.217 | 0.023 |

Table 1. Comparison between PSMNet and the proposed method in terms of disparity accuracy.

In addition to the disparity accuracy, the execution speed of the proposed dense stereo vision system is also quantified to evaluate the overall system's performance. Owing to the fact that the image size and disparity range are not constant among different datasets, a general way of evaluating the performance in terms of processing speed is to measure millions of disparity evaluations per second [27]:

$$Mde/s = \frac{u_{\max} v_{\max} d_{\max}}{t} \times 10^{-6}, \tag{36}$$

where the resolution of the disparity map is $u_{\max} \times v_{\max}$, $d_{\max}$ is the maximum disparity value, and $t$ is the processing time in seconds. The runtime of the proposed dense stereo vision system on the Jetson TX2 GPU is approximately 152.889 ms, and the resolution of the disparity map is $695 \times 361$. Therefore, the value of $Mde/s$ is 49.231, which is much higher than most stereo vision systems implemented on powerful graphics cards.

#### 4.4.2 Roll Angle Estimation

In the experiments, we select a range of $\lambda^{(0)}$ and record the number of iterations that (25) takes to converge to the minimum. It is shown that $\lambda^{(0)} = 10$ is the optimum value when the threshold $\delta_\psi$ is set to $\frac{\pi}{1.8\times10^6}$ rad ($0.0001°$).

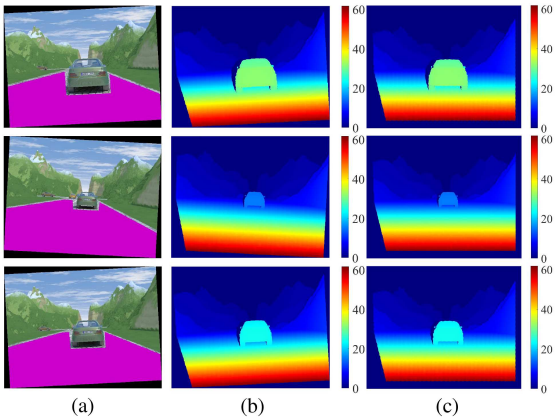


Figure 6. Examples of the roll angle estimation experiments; (a) reference images, the areas in magenta are the manually selected road regions; (b) original disparity maps; (c) disparity maps rotated around the estimated roll angles.

Furthermore, a synthesised stereo dataset from EISATS[6] [29,31] is used to quantify the accuracy of the proposed roll angle estimation algorithm. The roll angle of each image in this dataset is perfectly zero. Therefore, we manually rotate the disparity maps around a given angle, and then estimate the roll angles from the rotated disparity maps. Examples of the roll angle estimation experiments are shown in Figure 6, where it can be observed that the effects due to image rotation are effectively corrected. When $\delta_\psi$ is set to $\frac{\pi}{1.8\times10^6}$ rad, the average difference $\Delta\theta$ between the actual and estimated roll angles is approximately $0.012$ rad. The runtime of the proposed roll angle estimation on the Jetson TX2 GPU is approximately 7.842 ms.

#### 4.4.3 Disparity Transformation

In [5], Fan *et al.* published three road datasets containing various types of road damages, such as potholes and cracks. Therefore, we first use their datasets to qualitatively evaluate the performance of the proposed disparity transformation algorithm. Examples of the transformed disparity maps are illustrated in Figure 7, where it can be observed that the disparities of the road surface have similar values, while their values differ greatly from those of the road damages. This fact enables the damaged road areas to be easily recognised from the transformed disparity maps.

The KITTI stereo datasets are further utilised to evaluate the performance of disparity transformation. Examples of the KITTI stereo datasets are shown in Figure 8. To quantify the accuracy of the transformed disparities, we compute the standard deviation $\sigma_d$ of the transformed disparity values as

[6] https://ccv.wordpress.fos.auckland.ac.nz/eisats/set-2/

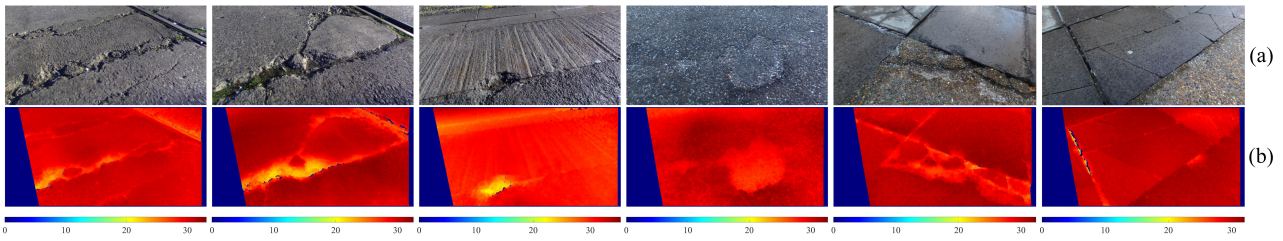


Figure 7. Examples of the disparity transformation experiments; (a) reference images; (b) transformed disparity maps.

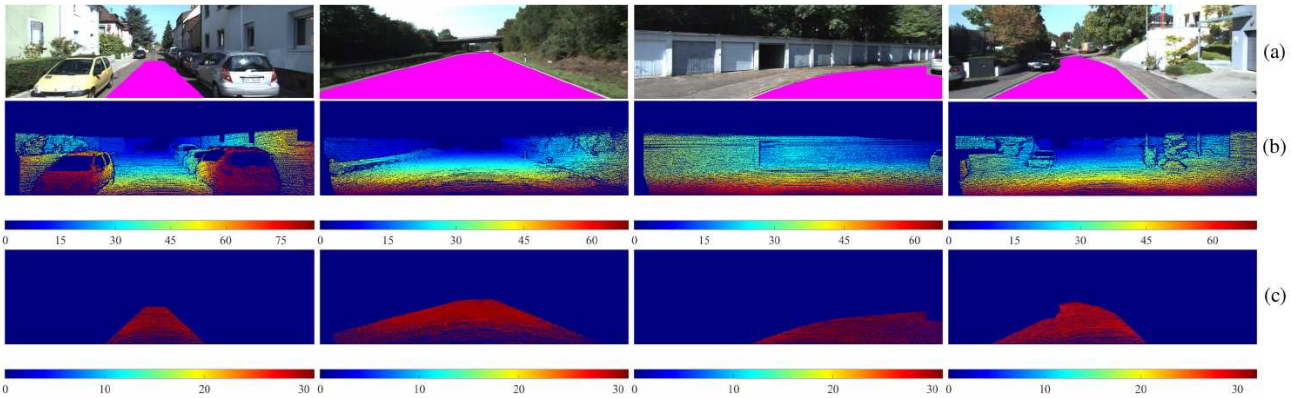


Figure 8. Disparity transformation experimental results of the KITTI stereo datasets; (a) reference images, where the areas in magenta are the manually selected road regions; (b) ground truth disparity maps; (c) transformed disparity maps.

follows:

$$\sigma_d = \sqrt{\frac{1}{m}\left\|\mathbf{s} - \frac{\mathbf{s}^\top \mathbf{1}_m}{m}\right\|_2^2}, \tag{37}$$

where $\mathbf{s} = [\ell'(\mathbf{p}_0), \ell'(\mathbf{p}_1), \ldots, \ell'(\mathbf{p}_{m-1})]^\top$ stores the transformed disparity values. The average $\sigma_d$ value of the KITTI stereo datasets is 0.519 pixels. However, if the image rotation effects caused by the non-zero roll angle are not eliminated, the average $\sigma_d$ value becomes 0.861 pixels. The runtime of the disparity transformation on the Jetson TX2 GPU is around 1.541 ms.

## 5. Conclusion and Future Work

This paper presented a robust dense stereo vision system embedded in a DJI Matrice 100 UAV for road condition assessment. The perspective transformation greatly improved the disparity accuracy and reduced the algorithm computational complexity, while the disparity transformation algorithm enabled the UAV to estimate roll angles from disparity maps. The damaged road areas became highly distinguishable in the transformed disparity maps, and this can provide new opportunities for UAV-based road damage inspection. The proposed system was implemented with CUDA on a Jetson TX2 GPU, and real-time performance was achieved.

In the future, we plan to use the obtained disparity maps to estimate the flight trajectory of the UAV and reconstruct the 3D maps using the state-of-the-art simultaneous localisation and mapping (SLAM) algorithms.

## 6. Acknowledgment

This work is supported by grants from the Research Grants Council of the Hong Kong SAR Government, China (No. 11210017 and No. 21202816) awarded to Prof. Ming Liu. This work is also supported by grants from the Shenzhen Science, Technology and Innovation Commission, JCYJ20170818153518789, and National Natural Science Foundation of China (No. 61603376) awarded to Dr. Lujia Wang.

## References

[1] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on pattern analysis and machine intelligence*, 23(11):1222–1239, Nov. 2001.

[2] J.-R. Chang and Y.-S. Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018.

[3] L. Cruz, D. Lucio, and L. Velho. Kinect and rgbd images: Challenges and applications. In *Proc. Patterns and Images Tutorials 2012 25th SIBGRAPI Conf. Graphics*, pages 36–49, Aug. 2012.

[4] R. Fan. *Real-time computer stereo vision for automotive applications*. PhD thesis, University of Bristol, 2018.

[5] R. Fan, X. Ai, and N. Dahnoun. Road surface 3D reconstruction based on dense subpixel disparity map estimation. *IEEE Transactions on Image Processing*, PP(99):1, 2018.

[6] R. Fan, M. J. Bocus, and N. Dahnoun. A novel disparity transformation algorithm for road segmentation. *Information Processing Letters*, 140:18–24, 2018.

[7] R. Fan, Y. Liu, X. Yang, M. J. Bocus, N. Dahnoun, and S. Tancock. Real-time stereo vision for road surface 3-d reconstruction. In *2018 IEEE International Conference on Imaging Systems and Techniques (IST)*, pages 1–6. IEEE, 2018.

[8] W. Feng, W. Yundong, and Z. Qiang. Uav borne real-time road mapping system. In *2009 Joint Urban Remote Sensing Event*, pages 1–7. IEEE, 2009.

[9] A. Fox, B. V. Kumar, J. Chen, and F. Bai. Multi-lane pothole detection from crowdsourced undersampled vehicle sensor data. *IEEE Transactions on Mobile Computing*, 16(12):3417–3430, 2017.

[10] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 3354–3361, June 2012.

[11] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[12] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelli- gence*, 30(2):328–341, Feb. 2008.

[13] E. T. Ihler, J. W. F. Iii, and A. S. Willsky. Loopy belief propagation: Convergence and effects of message errors, 2005.

[14] M. R. Jahanshahi, F. Jazizadeh, S. F. Masri, and B. Becerik-Gerber. Unsupervised approach for autonomous pavement-defect detection and quantification using an inexpensive depth sensor. *Journal of Computing in Civil Engineering*, 27(6):743–754, 2012.

[15] T. Kim and S.-K. Ryu. Review and analysis of pothole detection methods. *Journal of Emerging Trends in Computing and Information Sciences*, 5(8):603–608, 2014.

[16] C. Koch and I. Brilakis. Pothole detection in asphalt pavement images. *Advanced Engineering Informatics*, 25(3):507–515, 2011.

[17] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth. A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. *Advanced Engineering Informatics*, 29(2):196–210, 2015.

[18] C. Koch, G. M. Jog, and I. Brilakis. Automated pothole distress assessment using asphalt pavement video data. *Journal of Computing in Civil Engineering*, 27(4):370–378, 2012.

[19] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[20] W. Luo, A. G. Schwing, and R. Urtasun. Efficient deep learning for stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5695–5703, 2016.

[21] S. Mathavan, K. Kamal, and M. Rahman. A review of three-dimensional imaging technologies for pavement distress detection and measurements. *IEEE Transactions on Intelligent Transportation Systems*, 16(5):2353–2362, 2015.

[22] M. Menze, C. Heipke, and A. Geiger. Joint 3d estimation of vehicles and scene flow. *ISPRS Workshop on Image Sequence Analysis (ISA)*, II-3/W5:427–434, 2015.

[23] M. G. Mozerov and J. van de Weijer. Accurate stereo matching by two-step energy minimization. 24:1153–1163, 2015.

[24] P. Pedregal. *Introduction to optimization*, volume 46. Springer Science & Business Media, 2006.

[25] E. Schnebele, B. F. Tanyu, G. Cervone, and N. Waters. Review of remote sensing methodologies for pavement management and assessment. *European Transport Research Review*, 7(2):1, Mar. 2015.

[26] Tappen and Freeman. Comparison of graph cuts with belief propagation for stereo, using identical mrf parameters. In *Proc. Ninth IEEE Int. Conf. Computer Vision*, pages 900–906 vol.2, Oct. 2003.

[27] B. Tippetts, D. J. Lee, K. Lillywhite, and J. Archibald. Review of stereo vision algorithms and their suitability for resource-limited systems. *Journal of Real-Time Image Processing*, 11(1):5–25, 2016.

[28] Y.-C. Tsai and A. Chatterjee. Pothole detection and classification using 3d technology and watershed method. *Journal of Computing in Civil Engineering*, 32(2):04017078, 2017.

[29] T. Vaudrey, C. Rabe, R. Klette, and J. Milburn. Differences between stereo and motion behaviour on synthetic and real-world stereo sequences. In *Image and Vision Computing New Zealand, 2008. IVCNZ 2008. 23rd International Conference*, pages 1–6. IEEE, 2008.

[30] P. Wang, Y. Hu, Y. Dai, and M. Tian. Asphalt pavement pothole detection and segmentation based on wavelet energy field. *Mathematical Problems in Engineering*, 2017, 2017.

[31] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke, and D. Cremers. Efficient dense scene flow from sparse or dense stereo data. In *European conference on computer vision*, pages 739–751. Springer, 2008.

[32] S. Zagoruyko and N. Komodakis. Learning to compare image patches via convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4353–4361, 2015.

[33] J. Zbontar and Y. LeCun. Computing the stereo matching cost with a convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1592–1599, 2015.

[34] C. Zhang. An uav-based photogrammetric mapping system for road condition assessment. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci*, 37:627–632, 2008.

[35] C. Zhang and A. Elaksher. An unmanned aerial vehicle-based imaging system for 3d measurement of unpaved road surface distresses 1. *Computer-Aided Civil and Infrastructure Engineering*, 27(2):118–129, 2012.

[36] C. Zhou, H. Zhang, X. Shen, and J. Jia. Unsupervised learning of stereo matching. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1567–1575, 2017.