Transactions of KSAE, Vol. 33, No. 5, pp.365-375 (May, 2025)

pISSN 1225-6382 / eISSN 2234-0149
DOI http://dx.doi.org/10.7467/KSAE.2025.33.5.365

특징 공간의 확장과 경량 CNN을 활용한 우천 시 도로 영상의 스테레오 매칭

송 진 규 · 이 준 웅*

전남대학교 산업공학과

Stereo Matching for Rainy Road Images Using Feature Space Expansion and Lightweight Convolutional Neural Network

JinGyu Song · JoonWoong Lee*

Department of Industrial Engineering, Chonnam National University, Gwangju 61186, Korea
(Received 2 January 2025 / Revised 5 February 2025 / Accepted 8 February 2025)

Abstract : Various ill-posed regions, such as repeated patterns and reflective surfaces, pose pressing challenges to accurate stereo matching(SM). In road images captured in rainy weather, these ill-posed regions are randomly scattered across the entire image, making accurate matching even more difficult. To address this, this study proposed a novel method to achieve accurate sparse SM(SSM) even under challenging conditions caused by rain. The suggested method include the following steps: (1) constructing an image pyramid to capture global structures and fine details; (2) generating multiscale feature maps for stereo images at each resolution using a lightweight neural network; (3) enhancing the flexibility of matching cost calculation by splitting each feature map into multiple groups along the channel axis; and (4) filtering the matching cost using a cascaded lightweight network. The featured method was validated using the GYLane and KITTI datasets, demonstrating its effectiveness in addressing complicated conditions. The findings indicate that the recommended method outperforms existing approaches by more than 10 % in the D1 metric and by more than 2 points in the EPE metric. The code is available at https://github.com/sjg918/raindisp.

Key words : Autonomous driving(자율주행), Convolution neural network(합성곱 신경망), Feature space expansion(특징 공간의 확장), Matching cost calculation(매칭 비용 계산), Sparse stereo matching(희소 스테레오 매칭)

1. 서 론

차량에서 스테레오 카메라는 객체나 차로(Lane) 정보 검출 등의 분야에 사용되고 있다.[1-3] 스테레오 매칭(Stereo Matching, SM)은 시차(Disparity)를 추정하여 이러한 분야에 제공한다.[4-6] 이때 영상에 잡음이 존재하면 SM의 정확도가 제한될 수 있지만, 최근 합성곱 신경망(Convolutional Neural Network CNN)을 적용하여 정확도가 상당히 향상되었다.

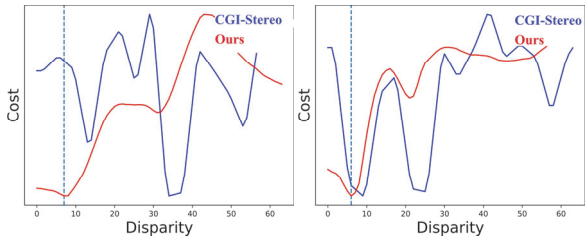
대부분의 CNN 기반의 SM 방법들[7-11]은 깊은(Deep) 2차원(2D) CNN이 추출한 특징들을 사용하여 비용 볼륨들을 생성한 후, 이 볼륨들내의 비용을 3차원 합성곱(3DConv)을 이용하여 모으고(Aggregation), 이 모음 결과로 생성된 볼륨을 시차 회귀에 사용한다. 이러한 CNN 기반의 방법들은 반복 패턴 및 반사 표면과 같은 불완전(Ill-posed) 영역에 속한 픽셀들의 매칭도 CNN이 추출한 맥락(Context) 정보를 이용하여 정확히 수행한다. 그러나 이 논리는 불완전 영역이 영상의 일부에 국한되는 경우에는 성립하지만, 불완전 영역들이 영상 전체에 무작위로 나타나는 경우에는 성립되지 않는다. 즉, 우천 시 차량 내에서 촬영된 옥외의 도로 영상에는 빗방울들이나 노면상의 물 고임 등에 의해 야기된 불완전 영역들이 영상 전체에 무작위로 나타나므로 기존의 CNN 기반의 방법들은 그들의 장점을 발휘하기 어렵다.

Fig. 1은 우천 시 야간에 촬영된 도로 영상의 차선경계에서 추출한 픽셀들(Fig. 1(a)의 작은 노란색 원으로 표기됨)에 대해 CNN 기반의 방법들 중 하나인 CGI-Stereo[11]

*Corresponding author, E-mail: joonlee@chonnam.ac.kr

와 본 연구에서 제안한 방법을 사용하여 계산한 초기 매칭 비용의 분포를 보인 것이다. Fig. 1(b)에 표시된 녹색의 수직 점선은 실제(Ground Truth, GT) 시차의 위치를 표시한 것이다. CGI-Stereo로 생성한 비용 분포 곡선(Fig. 1(b)의 파랑색)에는 GT시차와 관련 없는 위치에 피크들이 나타났다. 따라서 CGI-Stereo는 시차 계산 시 오류 발생 가능성이 크다. 반면에 Fig. 1(b)의 빨간색 곡선은 본 연구에서 제안된 방법을 사용하여 얻은 것으로, GT시차 근방에서 비용이 낮다. 따라서 정확한 시차 추정 가능성이 높다.

(a) An image captured on a rainy night



(b) Initial matching cost distribution: left - for pixel 2, right - for pixel 3 (from image in (a))

Fig. 1 Initial matching cost calculation for pixels on unclear edges affected by noise

본 연구는 우천 시 주야에 도로에서 촬영한 스테레오 영상에서 차선의 경계처럼 밝기 기울기가 비교적 큰 곳에 있는 픽셀들의 시차를 정확히 추정할 수 있는 CNN을 제안한다. 우천 시 주야에 도로를 촬영한 스테레오 영상에서 모든 픽셀들을 대상으로 시차를 추정하는 연구는 찾아보기 어렵다. 우리는 기존의 방법들과 달리 CNN을 경량 구조로 설계하고, 이를 LCNN(Lightweight Convolutional Neural Network)이라 명명하였다. 우리가 설계한 LCNN은 우천 시 도로 영상에서 추출된 픽셀들의 시차 추정에 적합하며, 처리시간이 짧다는 장점이 있다.

우리는 다음에 제시한 세 어프로치들을 사용하여 우천 시 도로를 촬영한 스테레오 영상에 대해 정확하고 신속한 픽셀 매칭을 수행한다. 첫째, 특징 추출 네트워크(Feature Extraction Network, FEN)와 비용 필터링 네트워크(Cost Filtering Network, CFN)을 경량 구조로 설계한다. 특히 FEN은 기존의 복잡한 심층망과 달리 단지 두 층의 2D 합성곱 망으로 설계하여 영상의 미세한 특징 검출에 집중한다.

둘째, 비로 인해 잡음이 혼재된 영상에서 픽셀 매칭에 활용할 수 있는 특징들의 선택 폭을 넓히기 위해 특징 공학 기법을 이용하여 영상의 특징 공간을 확장한다. 이 과정은 고해상도의 원본 영상부터 원본 영상의 1/32까지 6단계 스케일로 이루어진 영상 피라미드 구축으로 시작된다. 기존 CNN 기반 방법들은 대개 심층망내에서 다운(Down) 샘플링과 업(Up) 샘플링을 통해 수용 영역(Receptive field)을 넓히고 맥락 정보가 추출된 특징맵(Feature Map, FM)을 생성하지만, 본 연구에서는 고해상도 영상은 세부(Detail) 정보를, 저해상도 영상은 공간적이고 구조적인 맥락 정보를 유지한다는 전제하에 영상 피라미드를 수동으로 구축한다. 그리고 스케일별로 좌우 영상을 경량 구조의 FEN에 입력하여 32개 채널의 좌우 FM을 얻는다. 우리는 이렇게 얻은 여섯 쌍의 좌우 FM들에 대해 GWCNet[12]에서 제안된 방법과 유사하게 FM들을 채널 축을 따라 동일 크기의 여러 그룹들로 나눈 후, 이 그룹들 간의 매칭 비용을 계산한다. 이와 같은 시도를 통해 특징 공간을 확장하여 비로 인한 잡음 효과를 극복하고 정확한 매칭 가능성을 높인다.

셋째, 기존 CNN 기반의 SM 방법들이 비용 모음에 사용한 시간 소요가 크고, 비로 인해 유발된 잡음의 효과를 주위 픽셀들에 전파할 수 있는 3DConv 대신에 처리 시간을 획기적으로 줄인 매우 간단하고 가벼운 캐스케이드(Cascade) 구조의 CFN을 제안하여 매칭 비용을 필터링한다.

기존의 규칙 기반의 SM 방법들[13,14]은 다양한 규칙을 사용하여 픽셀간 매칭 비용 계산을 위해 수제(Hand-crafted) 특징을 추출하고, 지원 영역(Support region)을 결정한다. 그러나 우천 시 촬영한 영상에는 많은 잡음이 혼재되어 있기 때문에 매칭을 위해 사전에 설정된 규칙들을 적극적으로 적용하기는 어렵다. 반면에 LCNN은 앞서 언급한 세 가지 접근법을 기반으로 이러한 잡음 문제를 효과적으로 극복한다.

우리는 전남 광양에서 우천 시 주야에 촬영한 400장의 영상을 사용하여 기존 CNN 기반 SM 방법들과 LCNN의 성능 비교 실험을 수행했다. 400장 영상에는 주야 각각 촬영한 200장의 영상이 포함된다. 실험 결과 LCNN이 최신 CNN 기반 SM 방법들보다 성능이 우수했다. 우리는 또한 KITTI 데이터 집합[15]에 대해서도 비교 실험을 수행했다. 여기서 강조할 점은 우천시 촬영한 영상으로 제안

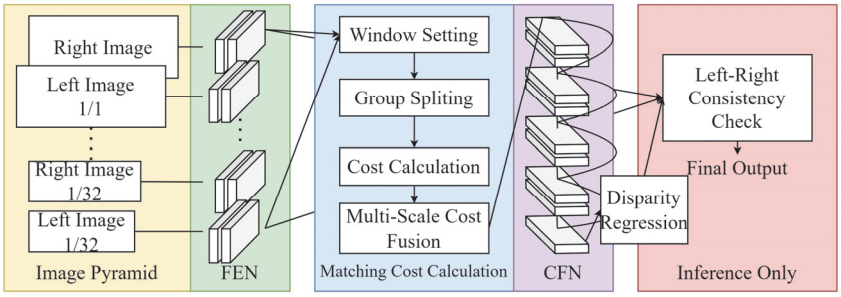


Fig. 2 Schematic architecture of our proposed method

된 방법을 훈련하지 않았다는 것이다.

이어지는 논문의 구성은 다음과 같다. 2장에서 관련 연구를 소개하고, 3장에서 본 연구에서 우천 시 촬영한 영상의 SM을 위해 제안한 방법을 설명한다. 4장에서는 실험 결과를 제시하고, 5장에서 논문을 마무리한다.

2. 관련 연구

이 장에서는 대개 4단계로 구성된 SM 방법의 각 단계를 설명하고, 기존 방법과 본 연구에서 제안된 방법의 차이점을 설명한다. CNN 기반의 SM의 첫 단계는 특징 추출이다. 일반적으로 FEN은 팽창 합성곱의 심층 구조를 사용하여 풍부한 문맥 정보를 추출한다. 또한 공간 피라미드 풀링(Pooling) 모듈을 사용하여 수용 영역을 넓힌다.[7] 최근 일부 방법에서는 높은 계산 비용이 발생하는 구조 대신 MobileNetV2[17]를 특징 추출에 사용했다. MobileNetV2는 더 적은 매개변수로 과적합을 억제하는 동시에 역잔차 블록을 사용하여 계산 비용을 줄인다. 그러나 기존 FEN은 여전히 계산량이 많다. 이와 대조적으로 우리가 제안한 FEN은 초경량 구조로 설계되었다.

두 번째 단계는 비용 볼륨을 구축하는 것으로, CNN 기반의 방법들은 일반적으로 좌우 특징 간의 상관 관계를 계산하거나 좌우 특징을 직접 연결하여 비용 볼륨을 구축한다. 특징 간의 상관관계를 사용하면 기하학적 구조 정보를 얻을 수 있는 반면, 연결된 볼륨은 더 자세한 정보를 제공한다.[10] 그러나 우천 시 촬영한 도로 영상 전체에 무작위로 존재하는 잡음은 특징 상관 관계로부터 얻은 기하학적 구조 정보를 무의미하게 만들 수 있다. 특징 간의 상관관계를 계산하거나 직접 연결하는 대신, 제안된 방법은 특징 간의 절대 차이를 계산한다. 최근, 비용 볼륨의 크기를 줄이고 매칭 시간을 단축하기 위한 Coarse-to-fine 방법[18]이 제안되었다. 이 방법은 공간 해상도를 유지하면서 시차 평면의 수를 조정하여 비용 볼륨의 크기를 제어한다.

세 번째 단계는 2D 또는 3DConv로 수행하는 비용 모음이다. 대개 모래시계 구조[7]의 네트워크가 모음 과정에서 의미론적(Semantic) 정보 추출을 위해 채택된다. 그런데 3DConv를 사용하여 우천 시 촬영한 도로 영상에서 픽셀들에 대한 매칭 비용을 모을 경우 역효과가 나타날 수 있다. 3Dconv 과정은 고려 중인 기준 픽셀 주위의 픽셀들의 비용을 모으는 것인데, 이때 영상 전체에 무작위로 존재하는 불완전 영역의 잡음 픽셀들의 비용이 이 모음에 반영된다. 결과적으로 3DConv는 필연적으로 잡음이 내재된 비용을 계속 누적하고 전파한다. 또한 최근 시간이 많이 드는 방법 대신에 ConvGRU(Convolutional Gated Recurrent Unit)[16]를 사용하는 방법이 제안되었다. 이 방법은 ConvGRU를 사용하여 비용 볼륨에서 비용을 반복적으로 샘플링하고 증분적 불일치 정제를 수행한다.

네 번째 단계는 Softargmin 함수를 사용하여 비용 모음을 통해 구축된 비용 볼륨을 정규화하고 이를 시차맵으로 회귀하는 과정이다. 제안된 방법도 시차 회귀는 이와 동일한 과정을 따른다.

3. 제안된 방법

이 장은 본 연구에서 제안한 방법을 설명한다. Fig. 2는 제안된 방법의 개략적인 구조를 보인 것이다. 여기에는 FEN, 다중 스케일 및 그룹별 비용 계산, CFN, 가려진 영역을 구분하기 위한 좌우 일관성 검사 등이 포함된다.

3.1 특징 추출

좌우 영상 사이에서 대응점을 탐색하려면 영상으로부터 특징 추출이 필요하다. 이를 위해 우리는 두 개의 2D 합성곱 층으로만 구성된 초경량 구조의 FEN을 설계했다.

특징 추출 과정에서 우리는 우천 시 촬영한 도로 영상에서 영상 전체에 무작위로 나타난 불완전 영역들로 인한 SM의 어려움 해결을 위해 여러 시도를 한다. 그 첫 시도로 입력 영상의 다운샘플링을 통해 영상 피라미드를 만든다. 우리는 원 영상의 해상도 H × W(높이 × 너비)에서 시작하여 해상도를 1/2씩 줄여 가면서 다음과 같이 6

가지 스케일의 영상을 생성한다. H × W, 1/2H × 1/2W, 1/4H × 1/4W, 1/8H × 1/8W, 1/16H × 1/16W 및 1/32H × 1/32W가 그것들이다. 이 스케일 결정은 실험적으로 이루어졌다.

우리는 6가지 스케일의 영상 각각을 FEN에 입력하여 해당 스케일의 FM을 생성한다. 즉 해상도별로 스테레오 영상이 제공되며, Fig. 2의 좌측에 표시된 것처럼 동일한 구조의 FEN 12개를 사용하여 입력된 6가지 스케일의 좌우 영상마다 각각의 좌우 FM을 생성한다. 이때, 동일한 스케일 내의 좌우 영상 쌍에 사용된 FEN 쌍은 가중치를 공유하지만, 다른 스케일의 영상에 적용되는 FEN 쌍과는 가중치를 공유하지 않는다.

Table 1 Design specification of the FEN

Layer name	Setting	Output dimension
Conv1-1	7 × 7, bn, act	H × W × 32
Conv1-2	1 × 1	H × W × 32
Conv2-1	7 × 7, bn, act	1/2H × 1/2W × 32
Conv2-2	1 × 1	1/2H × 1/2W × 32
…	…	
Conv6-1	7 × 7, bn, act	1/32H × 1/32W × 32
Conv6-2	1 × 1	1/32H × 1/32W × 32

bn: batch normalization, act: activation function

Table 1은 우리가 설계한 FEN의 사양을 제시한 것이다. 여기서 층 이름의 첫 숫자는 층에 입력된 영상의 스케일 인덱스를 나타내고, 하이픈 뒤의 숫자는 층 순서를 나타낸다. FEN의 첫 번째 층에서는 7 × 7 필터를 사용한 합성곱 결과에 배치 정규화[19]를 적용하고 Leaky ReLU[20]를 활성화 함수로 사용한다. 두 번째 층에서는 1 × 1 필터로 합성곱만 수행한다. 각 FEN에서 출력되는 FM의 스케일은 입력 영상의 스케일과 동일하고, 채널은 스케일과 관계없이 32이다. 여기에서 필터 크기나 FM의 채널수는 실험적으로 결정했다.

본 연구에서 제안된 픽셀 매칭 비용 계산 방법의 장점 하나가 우리가 설계한 FEN 대신에 MobileNetV2[17]나 PSMNet[7]에 적용된 FEN을 사용할 수 있다는 점이다. 즉, 우리가 제안한 아키텍처가 일종의 플랫폼 역할을 한다. 사용된 FEN에 따른 SM의 성능 비교 실험 결과는 단원 4.3에 제시된다.

3.2 비용 계산

본 연구에서는 좌우 영상 사이의 대응점 탐색에 Fig. 3의 상단부에 보인 바와 같이 슬라이딩 윈도우 방식[15]을 이용하여 매칭 비용을 계산한다. 이 과정에 FEN이 생성한 다중 해상도의 FM들을 사용한다. 대응점 탐색은 좌 영상을 기준으로 수행하는데, 이때 좌 영상의 픽셀 $p = (u, v)$를 기준으로 p에 대응하는 우 영상의 픽셀 $t = (u\text{-}d, v)$를 찾는다. 여기에서 d는 시차(Disparity)로서 0 ~ *Maxdisp*-1까지의 범위로 설정했다. *Maxdisp*은 기존 CNN 기반의 SM 방법들과 성능 비교를 위해 기존 방법들과 동일하게 192로 정했다.

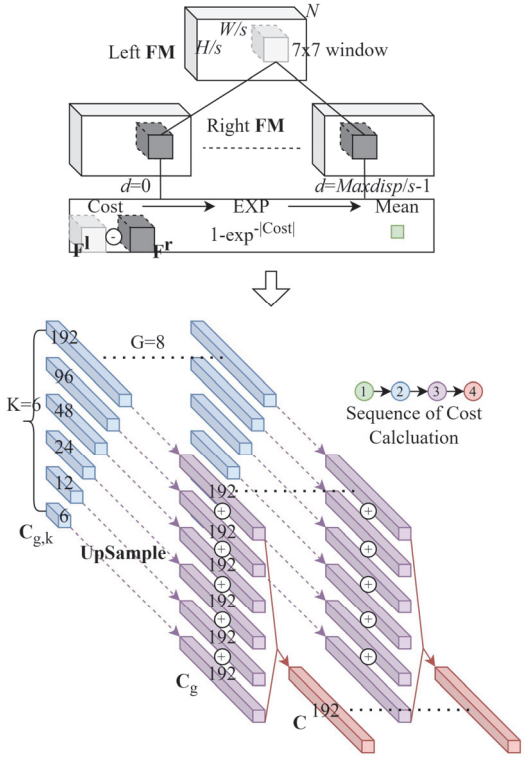


Fig. 3 Schematic description of multi-scale and group-wise matching cost calculation

우리는 좌 FM에는 p를, 우 FM에는 t를 중심으로 7 × 7 윈도우를 설정하고, 각 윈도우 내의 모든 픽셀들의 위치에서 해당 FM의 특징으로 p와 t의 매칭 여부를 판단할 비용을 계산한다. 여기에서 윈도우 크기 7 × 7은 실험적으로 정한 것이다. 모든 스케일의 FM들에 동일하게 7 × 7 윈도우를 적용하는데, FM의 스케일에 따라 p와 t의 좌표는 $p = (u/s, v/s)$, $t = ((u\text{-}d)/s, v/s)$가 된다. 따라서 좌우 FM에 설정된 윈도우 내의 픽셀들의 좌표는 식 (1)에 의해 결정된다.

$$
\begin{aligned}
&(u_l = \frac{u}{s} + i, v_l = \frac{v}{s} + i),\\
&(u_r = \frac{u-d}{s} + i, v_r = v_l),\\
&i = \{-3,-2,\ldots,2,3\},
\end{aligned}
\tag{1}
$$

여기에서 s는 FM의 스케일에 따른 다운 샘플링 상수로 1, 2, 4, 8, 16, 32 중 하나이고, 아래 첨자 l과 r은 각각 좌와 우를 나타낸다. 우측 FM에 대해 윈도우의 중심점 결정 시 d/s가 정수가 아닌 경우는 제외한다. 즉, 우측 FM에서 윈도우 이동 횟수가 해상도별로 달라진다. FM의 해상도가 H × W인 경우 *Maxdisp*번 이동하지만, H × W의 1/32인 경우 *Maxdisp*/32번 이동한다.

우리는 매칭 비용을 계산하기 위해 좌우 FM에서 추출한 특징을 각각 $\mathbf{F}^l$과 $\mathbf{F}^r$로 명명하고, 이들을 FM의 채널마다 생성한다. 이렇게 생성된 $\mathbf{F}^l$과 $\mathbf{F}^r$의 차원은 각각 $1 \times 7 \times 7 \times N(k)$와 $Maxdisp/s \times 7 \times 7 \times N(k)$이다. 여기에서 7×7은 윈도우 크기이며, k는 스케일 인덱스로서 1,…,6사이의 숫자이고, $N(k)$는 FM의 스케일에 따른 채널수를 나타낸다. 우리가 설계한 FEN은 FM의 공간 해상도와 관계없이 32 채널의 FM을 생성하지만, MobileNetV2나 PSMNet은 해상도에 따라 FM의 채널수가 다르다. 따라서 FM의 채널수 N을 해상도의 함수로 나타낸 것이다. 매칭 비용은 $\mathbf{F}^l$과 $\mathbf{F}^r$의 절대 차이로 계산한다. 기준이 되는 $\mathbf{F}^l$에 대해 $Maxdisp/s$개의 $\mathbf{F}^r$들 가운데 이 차이가 가장 작은 $\mathbf{F}^r$이 $\mathbf{F}^l$에 대응될 가능성이 높다. 우리는 이 차이 계산을 수월하게 하기 위해 $\mathbf{F}^l$을 $Maxdisp/s$개 복제해서 사용한다.

본 연구에서 우천 시 촬영된 좌우 영상의 대응점 탐색을 성공적으로 수행하기 위한 마지막 시도가 FM을 G개의 그룹으로 나누는 것인데, 우리는 $N(k)$를 8개의 그룹으로 나눈다. MobileNetV2나 PSMNet의 $N(k)$도 8개로 나눈다. 이 나눔은 대응점 탐색에 그룹별 매칭을 적용할 수 있는 기회를 제공한다. 여기에서 그룹수 8은 실험적으로 결정한 것이다. 우리가 설계한 FEN이 생성한 32채널의 FM을 8개 그룹으로 나누면 각 그룹은 4개의 채널로 이루어진다. 매칭 비용은 각 그룹에 속한 $\mathbf{F}^l$과 $\mathbf{F}^r$ 의 차이로 계산된다.

좌 영상의 픽셀 p에 대응하는 우 영상의 픽셀 탐색을 위한 비용 계산은 식 (2)에 의해 수행한다.

$$
\begin{aligned}
&\mathbf{C} = \{\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_8\}, \\
&\text{where} \\
&\mathbf{C}_g = \sum_{k=*}^{**} UpSample(\mathbf{C}_{g,k}), \\
&\mathbf{C}_{g,k} = \{c_{g,k,0}, c_{g,k,1}, \ldots, c_{g,k,(Maxdisp/s)-1}\}, \\
&c_{g,k,d} = \frac{\sum_{v=1}^{7}\sum_{u=1}^{7}\sum_{n=1}^{N(k)/8}(1-e^{-|Cost_{g,u,v,n,k,d}|})}{49 \times N(k)/8}, \\
&Cost_{g,u,v,n,k,d} = \mathbf{F}^l_{g,u,v,n,k} - \mathbf{F}^r_{g,u-d,v,n,k},
\end{aligned}
\tag{2}
$$

여기에서 아래 첨자 g, (u, v), n, d, s, k는 각각 그룹 인덱스, 윈도우 내의 픽셀 좌표, 그룹 내의 채널 인덱스, 시차, 다운 샘플링 상수, 스케일 인덱스를 나타낸다. 식 (2)에서 * 및 **는 본 연구에서 생성된 FM의 경우 1과 6인 반면 PSMNet에서 생성된 FM의 경우 모두 4이다. 숫자 49는 윈도우 내의 픽셀 수이다. 또한 $\mathbf{C}_{g,k}$는 선택된 그룹 g에 대한 $Maxdisp/s$차원의 비용인데, 우리는 $\mathbf{C}_{g,k}$에 선형 보간을 적용하여 *Maxdisp* 차원의 비용 $\mathbf{C}_g$를 생성한다. 식 (2)의 *UpSample*이 이 기능을 수행한다. 식 (2)의 비용 계산에 지수 함수를 도입한 것은 비정상적으로 큰 비용의 영향을 완화하기 위해서다. 식 (2)의 최종 결과인 $\mathbf{C}$는 $8 \times Maxdisp$ 행렬이다. Fig. 3은 식 (2)로 설명된 비용 계산 과정을 도식적으로 보인 것이다.

3.3 비용 필터링

우리는 Fig. 4에 보인 바와 같이 ResNet과 유사한 구조[21]로 설계한 CNN을 적용해 비용 필터링을 수행하는 새로운 방법을 제안한다. 이 CNN에서는 동일한 유형의 블록이 4번 반복되며, 각 블록은 2개의 2D 합성곱 층으로 구성된다. 이 CNN을 1장에서 언급한 대로 CFN으로 명명했다. 기존 방법들은 기준 픽셀의 이웃 픽셀들의 비용을 모으는 반면에 제안된 방법은 그룹 간의 매칭 비용을 필터링한다.

비용 필터링은 초기 비용 $\mathbf{C}$가 CFN의 첫 블록에 입력됨으로 시작된다. 이때 $8 \times Maxdisp$ 행렬인 $\mathbf{C}$는 $Maxdisp \times 1 \times 8$의 볼륨으로 간주되고, $\mathbf{C}$에 대한 2D 합성곱은 CFN의 각 합성곱 층에서 수행된다. 모든 합성곱 층의 출력은 볼륨의 공간 차원인 $Maxdisp \times 1$을 유지한다.

Fig. 4에서 “bn”은 배치 정규화를 나타내고, “act”는 Leaky ReLU 함수를 나타낸다. 따라서 CFN의 각 층에서 계산된 합성곱 결과에 배치 정규화를 수행한 후 Leaky ReLU 함수를 적용한다. 이 과정은 마지막 층을 제외한

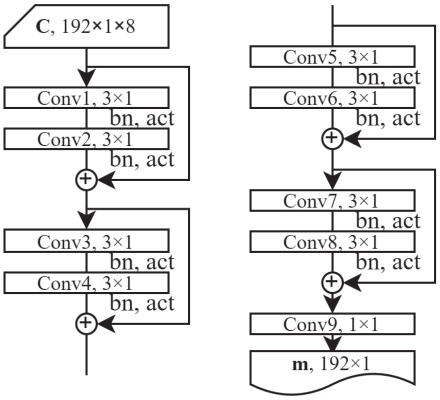


Fig. 4 CFN with a lightweight cascade structure

모든 층에 적용된다. 또한, 처음 8개 층에서 사용된 필터의 크기는 3 × 1이고, 이 8개 층의 출력은 채널수 8을 유지한다. 마지막 층에서 사용된 필터 크기는 1 × 1이고, 출력의 채널수는 1이다. 마지막 층의 역할은 각 시차 수준에 대한 8개 그룹의 매칭 비용을 통합하는 것이다. Fig. 4에서 ⊕는 덧셈을 의미한다. CFN의 최종 출력은 *Maxdisp* × 1 크기의 비용 벡터 **m**이다.

3.4 시차 회귀

우리는 CFN으로 얻은 *Maxdisp* × 1차원의 비용 **m**(*d*)로 시차 회귀를 수행한다. 이 회귀를 위해 **m**(*d*)에 -1을 곱한 후, CoEx[22]에서 제안된 상위 k-소프트-argmax(argmin) 시차 회귀의 수정된 버전인 상위 2개 값을 선택하고, 이들에 대해 소프트맥스 연산 $\sigma(.)$을 하여 시차를 예측한다.[9] 예측된 시차 d^*는 다음과 같이 두 시차 d_1과 d_2의 확률 가중합으로 계산된다.

$$d^* = d_1 \times \sigma(-m(d_1)) + d_2 \times \sigma(-m(d_2)), \quad (3)$$

여기서 $m(d_1)$은 **m**(*d*)에서 가장 작은 값이고, $m(d_2)$는 두 번째로 작은 값이다.

3.5 좌우 일관성 체크

시차 추론 시 좌우 일관성 체크(Left-Right Consistency Check, LRCC)는 가려진 영역들로 인해 발생할 수 있는 매칭 오류 식별을 위해 수행된다. 좌 영상의 기준 픽셀 $p = (u,v)$에 대응되는 우 영상의 픽셀이 t'으로 탐색된 경우, LRCC는 역으로 t'을 기준으로 t'에 대응되는 좌 영상의 픽셀이 p인지 여부를 확인한다.

LRCC도 식 (2)와 (3)을 이용하여 구현되는데, LRCC에서는 7 × 7 윈도우가 우측으로 이동된다는 점만 다르다. 식 (3)에 의해 예측된 시차가 d^*일 경우 t'의 좌표는 (u-d^*, v)이다. 따라서 LRCC 구현 시 좌영상에서의 윈도우 슬라이딩은 (u-d^*, v)에서 시작하여 ((u-d^*) + (*Maxdisp*/*s*-1), v)까지 이동한다. 여기에서 파라미터 *Maxdisp*과 *s*는 3.2절을 참고하기 바란다. 본 연구에서 t'에 대응하는 픽셀이 u±3의 범위 내에 있으면 t'을 p에 대응하는 픽셀로 간주하고, 그렇지 않으면 시차 예측이 실패한 것으로 간주한다. 여기에서 LRCC에 의한 시차 예측 성공 여부의 기준은 기존 연구[4]를 따른 것이다.

4. 실험결과

이 장에서는 본 연구에서 제안된 CNN의 훈련에 관한 세부 사항, 제안된 방법의 평가를 위한 데이터 준비, 평가에 사용된 척도를 제시한다. 또한 제안된 방법의 성능을 기존 CNN 기반의 방법들과 비교하며, 다양한 절제 연구의 결과도 제공한다.

4.1 훈련 세부사항

제안된 CNN의 훈련은 이전에 확립된 절차[7-11]를 따랐다. 먼저 960 × 540 크기의 35,454개의 훈련 영상과 4,370개의 평가 영상이 포함된 대규모 합성 데이터 집합인 Sceneflow 데이터 집합[23]를 사용하여 사전 훈련을 수행했다. 이 훈련에는 비대칭 색채 증강(Augmentation)[24]이 영상 비대칭성에 효과적으로 대처하기 위해 적용되었다. 증강에는 대상 및 참조 영상에 서로 다른 밝기([0.5, 2]), 감마([0.8, 1.2]), 대비([0.8, 1.2]) 및 채도([0, 1.4]) 조정을 무작위로 적용하는 것이 포함되었다. 또한, 무작위 자르기 증강도 훈련 중에 수행되었으며, 무작위로 잘린 이미지의 크기는 512 × 256이었다. 제안된 CNN은 배치 크기 4로 총 200에포크(epoch) 동안 학습되었다. 학습률은 처음에 0.001로 설정한 다음 80, 120, 160, 180에포크 후에 차례로 절반으로 줄였다. 학습은 단일 NVIDIA RTX 2080 Ti로 수행되었고, 최적화를 위해 Adam 옵티마이저[25]가 사용되었는데 이때 하이퍼 파라미터는 $\beta_1 = 0.9$ 및 $\beta_2 = 0.999$로 설정되었다.

이어서 실제 도로가 촬영된 영상 집합인 KITTI 2012와 KITTI 2015[17]를 사용하여 CNN을 훈련했다. KITTI 집합의 영상 해상도는 다양하지만 1248 × 384 미만이다. Sceneflow 데이터 집합에 사용된 것과 동일한 데이터 증강 기술이 적용되었고, 배치 크기가 4인 미니 배치로 총 600 에포크 훈련이 되었다. Sceneflow 데이터 집합에 의한 사전 훈련 때와 동일하게 초기 학습률은 0.001로 설정되었고, 200, 400, 500, 550 에포크 마다 순차적으로 절반씩 줄였다. 하드웨어 및 Adam 최적화 설정도 사전 훈련과 동일했다.

앞서 언급한 두 학습 모두 시차 추론의 대상 픽셀들이 필요하다. 이러한 픽셀들의 제공을 위해 다음과 같이 세 가지 방법을 사용했다. 첫 번째 방법은 3 × 3 Sobel 에지 연산자를 사용하여 밝기 영상의 기울기를 추출하고, 이 기울기의 크기가 120보다 큰 픽셀을 추출하는 것이다. 두 번째 방법은 영상 전체에서 완전히 무작위로 필요한 픽셀 수를 선택하는 것이다. 세 번째 방법은 첫 번째 방법과 두 번째 방법을 절반씩 조합하여 필요한 픽셀 수를 확보하는 것이다.

우리는 학습에 사용한 영상마다 64개의 픽셀을 추출하여 학습을 수행했다. 첫 방법을 통해 얻은 픽셀 수가 64개에 미치지 못하면 두 번째 방법을 사용하여 나머지 픽셀을 확보했다. 여기에서 언급한 픽셀 수 64는 실험적

으로 결정한 것이다.

우리는 이 세 가지 방법 각각을 사용하여 학습을 독립적으로 수행했고, 그 학습 결과를 저장했다. 이후 평가 과정에서 세 번째 방법이 가장 좋은 시차 추론 결과를 낳았다. 따라서 이 세 번째 방법을 선택하여 시차 추론을 수행할 픽셀들을 추출했다. 이 실험 결과는 단원 4.4에 제시하였다.

4.2 평가용 집합 GYLane과 KITTILane 구축

우천 시 도로 환경이 촬영된 영상이 포함된 공개 데이터 집합은 많지 않다. 따라서 우리는 제안된 방법의 평가를 위해 차량 내부에 ZED 2i 스테레오 카메라를 설치하고 한국 광양에서 우천 시 주행하면서 촬영한 영상들로 평가 데이터 집합을 구성했다. 주간 및 야간 영상 각 200개씩 총 400개의 영상을 선택했다. 이 영상들의 해상도는 1920 × 448이다. 이 영상들로부터 GT 시차 데이터 확보를 위해 영상에서 차선 경계 픽셀을 직접 선택하고, 해당 픽셀의 시차를 측정했다. 측정된 GT 시차에는 차선 경계 픽셀 외에도 횡단보도의 경계 픽셀들도 포함되었다. 이렇게 확보한 데이터 집합을 GYLane이라 명명했다.

또한, 제안된 방법의 평가에 KITTI 2015 데이터 집합의 200개 영상도 사용했다. 이 영상들의 차선 경계 픽셀은 차선 경계 픽셀 감지 방법[26]을 적용하여 추출했다. 그러나 추출된 차선 경계 픽셀들 중 KITTI에서 제공된 GT 시차맵에 시차가 존재하지 않거나 시차가 0.1 미만인 경우는 평가 대상에서 제외했다. 이렇게 구축한 집합을 KITTILane이라 명명했다.

4.3 FEN에 대한 SM 성능

제안된 FEN의 우수성 검증을 위해 KITTILane과 GYLane 집합을 사용하여 실험을 수행하였다. Table 2는 이 실험 결과를 보인 것이다. 우리가 제안한 FEN은 기존의 다른 두 방법들의 FEN에 비해 초경량 구조이지만, GYLane 집합으로 수행한 실험결과에서 SM 결과가 훨씬 좋았다. 반면에 KITTILane 집합의 결과에서는 제안된 FEN의 D1 오류[20]는 MobileNetV2보다 높지만 EPE 오류[20]는 가장 낮았다.

Table 2 SM results for different FENs

Method	KITTILane		GYLane	
	D1 (%)	EPE	D1(%)	EPE
PSMNet FEN[7]	0.40	0.693	10.84	4.871
MobileNetV2[17]	**0.24**	0.587	6.83	2.124
Ours	0.37	**0.586**	**3.86**	**0.850**

4.4 매칭 대상 선정 방법에 대한 SM 성능 비교

단원 4.1에서 제안된 CNN의 학습 시, 시차 추론을 위한 대상 픽셀 추출에 세 가지 방법을 사용했다고 설명했다. 이 단원에서는 KITTILane과 GYLane을 사용하여 수행한 실험 결과로부터 그 세 가지 방법들 중 어느 것이 가장 좋은 학습 결과를 낳았는지 확인한다. Table 3에 실험 결과를 제시하였다. Table 3의 첫 번째 열에 표기된 “Edge”, “Random” 및 “Random + Edge”는 각각 첫 번째, 두 번째 및 세 번째 방법을 나타낸다. 이 표에 제시된 결과는 셋째 방법으로 CNN을 학습한 것이 가장 좋은 결과를 낳았음을 보여 준다.

Table 3 SM performance according to matching target pixel extraction methods

Method	KITTILane		GYLane	
	D1 (%)	EPE	D1 (%)	EPE
Edge	0.54	0.630	3.98	1.384
Random	0.39	0.600	4.14	1.005
Random + edge	**0.37**	**0.586**	**3.86**	**0.850**

4.5 절제연구

우리가 제안한 그룹별 비용 계산, CFN, LRCC 등을 검증하기 위해 KITTILane 및 GYLane을 사용하여 절제 연구를 수행했다. Table 4는 이 절제 연구에 대한 실험 결과를 제시한 것이다.

Table 4의 첫 번째 열의 마지막 행에 있는 “Original”은 제안된 접근 방식들이 모두 적용된 경우를 나타낸다. “woGroup”은 단원 3.2에서 설명한 그룹별 비용 계산이 적용되지 않은 경우를 나타낸다. “woCFN”은 단원 3.3에서 설명한 CFN이 적용되지 않은 경우를 나타낸다. “Single Scale FEN”은 단원 3.1에서 설명한 다중 스케일 영상이 아닌 원본 영상만 적용되고, 이 원본 영상에 FEN 쌍이 적용된 경우를 나타낸다. “woLRCC”는 단원 3.5에서 설명한 LRCC가 적용되지 않은 경우를 나타낸다.

Table 4 Results of ablation studies

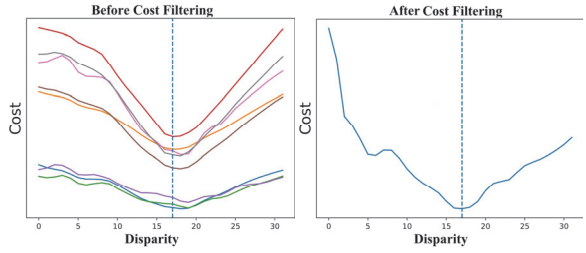
Method	KITTILane		GYLane	
	D1 (%)	EPE	D1 (%)	EPE
woGroup	0.52	0.659	4.66	0.844
woCFN	0.50	0.658	4.29	1.574
Single scale FEN	0.52	0.664	10.53	2.260
woLRCC	0.64	0.630	5.21	1.415
Original	**0.37**	**0.586**	**3.86**	**0.850**

Table 4에 제시된 결과는 KITTILane 및 GYLane 모두 우리가 적용했던 접근 방식들이 SM의 성능을 향상시키고 있음을 말해 주고 있다.

4.6 CFN의 효과

Fig. 5는 CFN의 효과를 보인 예다. Fig. 5(a)에서 화살표가 가리킨 빨간색 점이 시차가 추론된 픽셀을 나타낸다. Fig. 5(b)에 각각 다른 색상으로 표시된 그래프 8개는 단원 3.2에서 설명한 8개 그룹의 매칭 비용을 보인 것으로서 이들은 식 (2)를 사용하여 얻었으며, CFN의 첫 번째 층에 입력된 8×192 비용 행렬 **C**를 나타낸다. Fig. 5(b) 및 Fig. 5(c)에 그려진 수직 점선은 시차 추론 대상 픽셀의 GT 시차를 나타낸다. Fig. 5(b)를 보면 8개 그룹 초기 비용 각각이 GT 시차에 해당하는 곳에서 최소인 것은 아님을 확인할 수 있다. Fig. 5(c)에 보인 그래프는 CFN의 마지막 층의 출력으로 필터링된 비용 **m**을 나타낸 것이다. 이 사례를 통해 CFN에 입력된 초기 비용 **C**와 CFN에서 출력된 비용 **m**을 비교하면 CFN은 필터링 과정을 거쳐 GT 시차에 해당하는 곳에서 비용이 최소가 되게 한다는 것을 알 수 있다.

(a) A pixel requiring disparity in a nighttime rainy image



(b) Initial matching costs (c) Filtered matching cost

Fig. 5 Effect of CFN

4.7 GYLane에 의한 제안된 방법과 다른 방법들의 비교

우리 방법과 다른 방법들 간의 SM 성능 비교를 위해 GYLane 집합에 속한 영상들을 주간 및 야간 영상으로 나누었다. 그런 다음 이 두 유형의 영상에 대해 별도의 실험을 수행했다.

Table 5와 Table 6은 각각 GYLane의 주간 및 야간 영상으로 수행한 실험 결과를 제시한 것이다. 이 결과를 보면 제안된 방법이 다른 방법들보다 훨씬 우수한 SM 성능을

Table 5 SM errors of the proposed and other methods on GYLane daytime images

Methods	D1 (%)	EPE
CGI-Stereo[11]	16.08	3.518
FastACVNetPlus[10]	22.58	7.096
PCW-Net[8]	25.90	10.130
IGEVStereo[9]	22.35	11.924
PSMNet[7]	38.14	9.234
Ours	**3.86**	**0.850**

Table 6 SM errors of the proposed and other methods on GYLane nighttime images

Methods	D1 (%)	EPE
CGI-Stereo[11]	21.14	3.289
FastACVNetPlus[10]	28.51	5.815
PCW-Net[8]	63.56	6.481
IGEVStereo[9]	24.64	8.580
PSMNet[7]	59.59	5.735
Ours	**3.49**	**0.851**

보임을 알 수 있다. 또한 주간 및 야간 영상으로 별도의 실험을 수행했지만, 두 결과는 유사했다.

Fig. 6은 Table 5와 6에 제시된 결과 가운데 일부를 실험에 사용된 영상을 통해 보인 것이다. Fig. 6에 사용된 영상은 좌우 스테레오 영상 가운데 우 영상이며, 우리의 방법과의 비교를 위해 선택된 타 방법은 CGI-Stereo[11]이다. 본 연구에는 두 가지 유형의 시차가 있다. 하나는 GT 시차 d^T이고, 다른 하나는 추론된 시차 d^*다. Fig. 6에서 노란색 점은 d^T에 의한 점이고, 청록색 점은 d^*에 의한 점이다. 앞서 본론에서 설명했듯이 시차 추론은 좌영상을 기준으로 수행했기 때문에 우영상에 d^T와 d^*를 표시하기 위해 좌영상의 기준 픽셀의 위치에서 수평으로 d^T와 d^*만큼 좌측으로 이동하여 얻은 점들이 노란색과 청록색 점이다. 따라서 두 점이 겹치면 시차추론이 잘 된 것이고, 겹치지 않으면 시차추론에 에러가 있는 것이다. Fig. 6(a)와 Fig. 6(b)는 각각 CGI-Stereo와 우리가 제안한 방법을 사용하여 얻은 결과를 보인 것으로 CGI-Stereo와 달리 우리 방법은 빗방울이나 어두움에 관계없이 강건한 SM이 이루어졌음을 알 수 있다.

4.8 KITTILane에 의한 제안된 방법과 다른 방법들의 비교

KITTI 집합을 구성하는 영상은 모두 맑은 날씨에 촬영되었다. 따라서 CNN 기반의 기존 SM 방법들은 우천

(a) Stereo matching results for CGIStereo

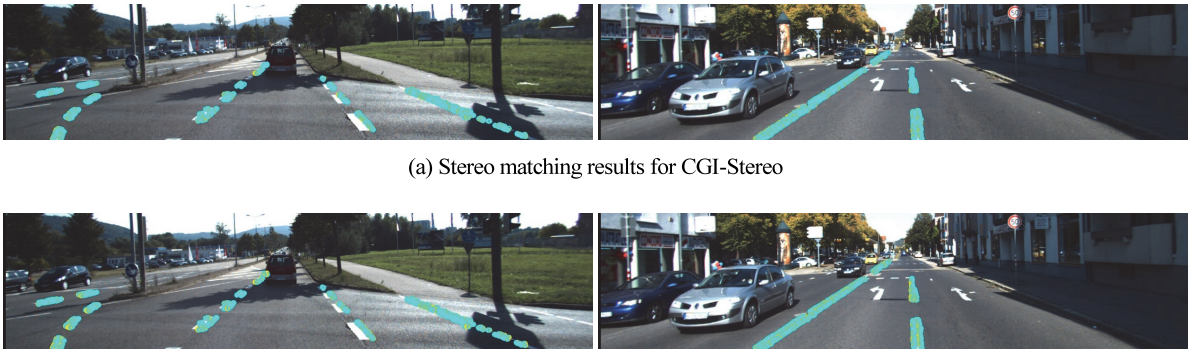
(b) Stereo matching results for our method

Fig. 6 Disparity estimation results for CGI-Stereo[11] and our method on GYLane

Table 7 SM errors of the proposed and other methods on KITTILane

Methods	D1 (%)	EPE
CGI-Stereo[11]	**0**	0.262
FastACVNetPlus[10]	**0**	0.245
PCW-Net[8]	0.1	0.426
IGEVStereo[9]	**0**	**0.219**
PSMNet[7]	0.3	1.083
Ours	0.37	0.856

시에 촬영된 영상과 달리 비로 인한 잡음이 없는 KITTI 영상에서 맥락 정보와 의미론적 정보를 충분히 활용할 수 있어서 좋은 성능이 기대된다. Table 7은 KITTILane을 사용하여 우리 방법과 다른 방법들 간의 SM 성능 비교를 위해 수행한 실험 결과를 제시한 것이다. Table 7의 D1 및 EPE 값은 Table 5와 Table 6의 해당 데이터를 얻을 때 사용된 것과 동일한 프로세스를 통해 얻었다. GYLane으로 행한 실험 결과와 달리 KITTILane으로 수행한 실험에서 Table 7에 제시된 바와 같이 우리 방법의 SM 성능이



(a) Stereo matching results for CGI-Stereo

(b) Stereo matching results for our method

Fig. 7 Disparity estimation results for CGI-Stereo[11] and our method on KITTILane

다른 방법들에 비해 상대적으로 낮았지만, 우리 방법도 PSMNet과의 D1 차이는 0.07에 불과하고 EPE는 더 양호한 결과를 낳았다.

Fig. 7은 Fig. 6에 보였던 것과 동일한 방식으로 Table 7의 결과를 얻는데 사용된 스테레오 영상들 가운데 일부의 우영상에 실험 결과를 표현해서 보인 것이다. Fig. 6에서와 같이 노란색과 청록색 점은 각각 GT 시차와 추론한 시차에 의해 생성한 점들을 나타낸다. Fig. 7을 육안으로 관찰하면 우리의 방법과 CGI-Stereo에 의한 결과 사이에 차이가 드러나지 않는다는 것을 알 수 있다.

4.9 처리시간 분석

우리가 제안한 방법은 1920 × 448 해상도의 영상에서 픽셀 100개의 시차를 추론하는데 22.53 ms 소요되었다. 이 시간은 좌우 영상의 특징을 추출하는 데에 8 ms, 비용 계산 및 필터링, 시차 회귀에 7.85 ms, 좌우 일관성 체크에 6.68 ms로 이루어진다.

더 많은 픽셀들의 매칭을 수행할 경우 더 긴 처리 시간이 필요하지만, 그렇다고 처리시간이 선형적으로 증가하는 것은 아니다. 제안된 방법은 1920 × 448 해상도의 영상에서 픽셀 500개의 시차를 추론하는데 22.53 ms의 5배인 112.65 ms가 아닌 68.85 ms가 소요되었고, 픽셀 2,500개의 시차를 추론하는 데는 307.89 ms 소요되었다.

5. 결 론

우천 시 촬영된 도로 영상에서 기존의 CNN 기반 SM 방법들은 영상 전체에 불완전 영역들이 무작위로 나타나기 때문에 문맥 정보를 효과적으로 활용할 수 없으며, 잡음 효과를 주변 픽셀에 전파하기 때문에 매칭 성능이 저하된다. 본 연구에서는 이러한 한계를 극복하기 위한 대안을 제안했다.

제안된 방법의 핵심은 SM에 활용할 특징 공간의 확대와 잡음 효과가 주위 픽셀로 전파되는 것을 막는 것이다. 이를 위해 우리는 영상 피라미드 구축, 초경량 FEN 설계, 다중 특징 맵 구축과 특징맵의 그룹화, 캐스케이드 구조의 CFN 설계 등을 시도하였다. 이러한 접근 방식들이 융합적으로 수행된 사례는 찾아보기 어려운 새로운 측면이 있다.

우리는 GYLane과 KITTILane으로 명명한 두 데이터 집합을 사용하여 수행한 실험에서 우리 방법이 비 오는 날씨와 맑은 날씨에 촬영한 영상 모두에서 양호한 SM 성능을 보임을 확인하였다. 양호한 날씨에 촬영된 영상으로 만든 KITTILane을 사용한 실험에서 제안된 방법이 기존 방법들을 능가하는 결과는 아니었지만, GYLane을 사용하여 수행한 실험에서 우리 방법은 다른 모든 방법들보다 성능이 뛰어났다. 이는 우천 시 영상과 양호한 날씨의 영상의 통계적 특성이 다름으로 인한 결과로 볼 수 있다.

후 기

이 연구는 2016년도 교육부 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(“RF-2016-NR017158”).

References

1) H. Badino, U. Franke and D. Pfeiffer, “The Stixel World - A Compact Medium Level Representation of the 3D-World,” 31st DAGM Symposium on Pattern Recognition, pp.51-60, 2009.
2) H. Hirschmuller, “Stereo Processing by Semiglobal Matching and Mutual Information,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.30, No.2, pp.328-341, 2008.
3) J. G. Song and J. W. Lee, “CNN-Based Object Detection and Distance Prediction for Autonomous Driving Using Stereo Images,” International Journal of Automotive Technology, Vol.24, No.3, pp.773-786, 2023.
4) G. Y. Song, S. I. Cho, D. Y. Kwak and J. W. Lee, “Accurate Dense Stereo Matching of Slanted Surfaces Using 2D Integral Images,” International Conference on Computer Vision Systems, pp.284-293, 2013.
5) E. Shibusawa, H. Sotodate and N. Fukushima, “Double-Direction Fisheye Plane Sweep Stereo for Road Surface 3D Reconstruction,” International Workshop on Advanced Imaging Technology (IWAIT), pp.117-122, 2023.
6) K. Y. Lee and J. W. Lee, “Intensity Gradients-Based Stereo Matching of Road Images,” Transactions of KSAE, Vol.11, No.1, pp.154-158, 2003.
7) J. R. Chang and Y. S. Chen, “Pyramid Stereo Matching Network,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.5410-5418, 2018.
8) Z. Shen, Y. Dai, X. Song, Z. Rao, D. Zhou and L. Zhang, “PCW-Net: Pyramid Combination and Warping Cost Volume for Stereo Matching,” European Conference on Computer Vision, pp.280-297, 2022.
9) G. Xu, X. Wang, X. Ding and X. Yang, “Iterative Geometry Encoding Volume for Stereo Matching,” Proceeding of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),

pp.21919-21928, 2023.
10) G. Xu, Y. Wang, J. Cheng, J. Tang and X. Yang, “Accurate and Efficient Stereo Matching via Attention Concatenation Volume,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.46, No.4, pp.2461-2474, 2024.
11) G. Xu, H. Zhou and X. Yang, “CGI-Stereo: Accurate and Real-Time Stereo Matching via Context and Geometry Interaction,” arXiv preprint arXiv: 2301.02789, 2023.
12) X. Guo, K. Yang, W. Yang, X. Wang and H. Li, “Group-Wise Correlation Stereo Network,” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp.3273-3282, 2019.
13) J. M. Park, G. Y. Song and J. W. Lee, “Shape-Indifferent Stereo Disparity Based on Disparity Gradient Estimation,” Image and Vision Computing, Vol.57, pp.102-113, 2017.
14) J. Lu, K. Zhang, G. Lafruit and F. Catthoor, “Real-Time Stereo Matching: A Cross-Based Local Approach,” IEEE International Conference on Acoustics, Speech, and Signal Processing, pp.733-736, 2009.
15) A. Geiger, P. Lenz and R. Urtasun, “Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite,” 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp.3354-3361, 2012.
16) L. Lipson, Z. Teed and J. Deng, “RAFT-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching,” 2021 International Conference on 3D Vision (3DV), pp.218-227, 2021.
17) M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp.4510-4520, 2018.
18) J. M. Park and J. W. Lee, “Improved Stereo Matching Accuracy Based on Selective Backpropagation and Extended Cost Volume,” International Journal of Control, Automation and Systems, Vol.20, No.6, pp.2043-2053, 2022.
19) S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” International Conference on Machine Learning, pp.448-456, 2015.
20) B. Xu, N. Wang, T. Chen and M. Li, “Empirical Evaluation of Rectified Activations in Convolutional Network,” arXiv preprint arXiv:1505.00853, 2015.
21) K. He, X. Zhang, S. Ren and J. Sun, “Deep Residual Learning for Image Recognition,” Proceedings of the IEEE Conference on Computer vision and Pattern Recognition, pp.770-778, 2016.
22) A. Bangunharcana, J. W. Cho, S. Lee, I. S. Kweon, K. S. Kim and S. Kim, “Correlate-and-Excite: Real-Time Stereo Matching via Guided Cost Volume Excitation,” 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp.3542-3548, 2021.
23) N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy and T. Brox, “A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp.4040-4048, 2016.
24) G. Yang, J. Manela, M. Happold and D. Ramanan, “Hierarchical Deep Stereo Matching on High-Resolution Images,” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.5515-5524, 2019.
25) D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” arXiv:1412.6980, 2014.
26) J. M. Park and J. W. Lee, “Lane Estimation by Particle-Filtering Combined with Likelihood Calculation of Line Boundaries and Motion Compensation,” Image and Vision Computing, Vol.79, pp.11-24, 2018.