
UNIVERSITATEA "ALEXANDRU IOAN CUZA" DIN IAŞI
FACULTATEA DE INFORMATICĂ



Lucrare de Licență

Percepția Adâncimii
din
Imagini Monocular

propusă de

Radu Manole

Sesiunea: Iulie, 2018

Coordonator științific
Prof. Dr. Anca Ignat

UNIVERSITATEA "ALEXANDRU IOAN CUZA" DIN IAŞI
FACULTATEA DE INFORMATICĂ

Lucrare de Licență

Percepția Adâncimii
din
Imagini Monocularare

propusă de

Radu Manole

Sesiunea: Iulie, 2018

Coordonator științific
Prof. Dr. Anca Ignat

Contents

Abstract	5
1 Introducere	7
1.1 Prezentare generală a temei și motivația	7
1.2 Date seturi de adâncime	8
2 Contribuții	11
3 Metodologii	13
3.1 Random Markov Field	13
3.2 Abordare non-parametrică	14
3.3 CNN cu două componente, <i>coarse(aspru)</i> global și <i>fine(fin)</i> local	16
3.4 Problemă de optimizare discretă-continuă	17
3.5 Regresie la caracteristicile adânci, și CRF-uri ierarhici	18
3.6 Învățare reziduală	21
3.7 Chen et al. [37] 2016	22
3.8 Învățarea Nesupervizată	23
3.8.1 Autoencoder	23
3.8.2 Reconstucție a imaginii în timpul antrenamentului	24
3.8.3 Zhou et al. 2017 antrenarea monoculară	26
3.9 Învățare semi-supervizată	28
3.10 Stereo vision de ultimă generație Deep3D	29
3.11 2017 perspectivă sintetizată(<i>view synthesis</i>) + potrivirea stereo (<i>stereo matching</i>)	29
3.12 Eroarea și exemple	31
4 Percepția adimii dintr-o singură imagine exemplu	33
4.1 Ground Plane Depth Coordinate System	33
4.1.1 Principiul de Formalre a Imaginei	34
4.2 Algoritmul de Generare a Sistemului de Coordonate de Adâncime	35
4.3 Algoritm de percepție a adâncimii în imagine	37
4.3.1 Segmentarea de Textură	38
4.3.2 Percepția adâncimii în imagine	38
5 Concluzie	41
Bibliography	43

Avizat,
Îndrumător de licență
Lect. dr. Ignat Anca
Data 29 Iunie 2018

DECLARAȚIE PRIVIND ORIGINALITATE ȘI RESPECTAREA DREPTURILOR DE AUTOR

Subsemnatul Radu Manole cu domiciliul în or. Rîșcani, raionul Rîșcani, Republica Moldova, născut la data de 3 martie 1994, identificat prin CNP 2004030006535, absolvant al Universității "Alexandru Ioan Cuza" din Iași, Facultatea de Informatică, specializarea Informatică, promoția 2016, declar pe propria răspundere, cunoscând consecințele falsului în declarații în sensul art. 326 din Noul Cod Penal și dispozițiile Legii Educației Naționale nr. 1/2011 art.143 al. 4 și 5 referitoare la plagiat, că lucrarea de licență cu titlul: "Percepția Adâncimii din Imagini Monocular" elaborată sub îndrumarea d-na Anca Ignat, pe care urmează să o susțină în fața comisiei este originală, îmi aparține și îmi asum conținutul său în întregime.

De asemenea, declar că sunt de acord ca lucrarea mea de licență să fie verificată prin orice modalitate legală pentru confirmarea originalității, consumând inclusiv introducerea conținutului său într-o bază de date în acest scop.

Am luat la cunoștință despre faptul că este interzisă comercializarea de lucrări științifice în vederea facilitării falsificării de către cumpărător a calității de autor al unei lucrări de licență, de diploma sau de disertație și în acest sens, declar pe proprie răspundere că lucrarea de față nu a fost copiată ci reprezintă rodul cercetării pe care am întreprins-o.

Iași, 29 iunie 2018

Absolvent Radu Manole

DECLARAȚIE DE CONSUMĂMÂNT

Prin prezenta declar că sunt de acord ca Lucrarea de licență cu titlul “Percepția Adâncimii din Imagini Monocular”, codul sursă al programelor și celealte conținuturi (grafice, multimedia, date de test etc.) care însotesc această lucrare să fie utilizate în cadrul Facultății de Informatică. De asemenea, sunt de acord ca Facultatea de Informatică de la Universitatea Alexandru Ioan Cuza Iași să utilizeze, modifice, reproducă și să distribue în scopuri necomerciale programele-calculator, format executabil și sursă, realizate de mine în cadrul prezentei lucrări de licență.

Iași, 29 iunie 2018

Absolvent Radu Manole

Abstract

Percepția adâncimii dintr-o singură imagine sau învățarea adâncimii dintr-o imagine monococulară, a fost o problemă cu interes din primele zile al domeniului de cercetare de recunoașterea imaginilor. Având la intrare doar o singură imagine 2D, este dorit ca metoda să extragă cit mai multă informație 3D, de obicei, fiind în formă de un tabel doi dimensional, cu fiecare celulă să corespundă cu pixelul respectiv din imagine. și valoarea reprezintă distanța de la obiectul din imagine pînă la cameră, sau pe scurt hartă de adâncime.

Creierul uman are puterea de a recunoaște și analiza imaginile cu ușurință. Pentru om este foarte usor să inteleaga structurile 3-d, distanța dintre ele și diferența dintre diferite obiecte. Însă pentru sistemele computațional de prelucrare a imaginilor, deducerea unui mesh 3D dintr-o imagine devine o problemă foarte complicată. Într-adevar, din perspectiva matematică producerea adâncimii 3-d este imposibila, din cauza amiguitatii, deoarece noi nu știm dacă imaginea este a unei picturi sau a unui mediu 3-d. Dar în practica oamenii pricep adâncimea cu o usurință remarcabilă, dîndu-i numai o singură imagine. Prin rezolvarea acestei probleme, dorința este că calculatorul să aibă aceiasi putere, sau aproximativ de recunoaștere a adâncimii într-o imagine. Pentru că această problemă este aplicată într-un număr mare de domenii importante, cum ar fi robotica, realitatea augmentată, reconstrucția 3D și conducerea automată a mașinelor etc.

Creierul omenesc rezolvă această problemă, cu ajutorul la numeroase indicii. Aceste indicii sunt de obicei grupate în patru categorii distincte: monoculare, stereo, mișcare parallax și de focus. Oamenii combina aceste indicii pentru a înțelege structura 3D a lumii. Majoritatea modelelor propuse în literatură de a rezolva această problemă, se bazează pe câteva indicii din aceste patru. Vizuala stereo este o problemă aparte la aceasta, deoarece constrângerea este de a avea doar o singură imagine a scenei, vizuala stereo nu poate fi direct folosită pentru rezolvarea problemei în cauză.

În ultima vreme cu avansarea mediului de informatică a inteligenței artificiale, sau creat diverse moduri probabilistice care au încercat să rezolve această problemă fără să utilizeze direct indicii de adâncime, ci să creeze o rețea care să înferne adâncimea prin antrenarea rețelei date. În această lucrare descriu diferite tipuri de și abordări din punct de vedere a învățării automate.

1 Introducere

În acest capitol descriu principalele motivații pentru a studia metodele de recunoaștere a adâncimii dintr-o imagine monoculară, și descrierea resurșelor pentru crearea metodelor probabilistice.

1.1 Prezentare generală a temei și motivația

Una dintre cele mai importante aplicații, este domeniul de robotică, senzorii de profunzime au devenit un instrument necesar pentru acest domeniu, datorită multitudinii de aplicații, ca de exemplu evitarea obstacolelor, navigație, localizarea și cartografierea mediului. Din această cauză senzorii cum ar fi Microsoft Kinect [38], sunt adesea folosiți pentru această problemă. Cu toate acestea, ele sunt foarte sensibile la lumina soarelui și sunt impractice pentru utilizarea în aer liber și, de asemenea, de regulă sunt foame și grele. Acest lucru le face foarte nepotrivite pentru a fi utilizate în aplicații de robotică de dimensiune mică, cum ar fi vehiculele micro-aeriene, care sunt constrânse din punct de vedere energetic și au capacitate foarte reduse de transport.

Pentru aceste aplicații, camerele stereo pasive sunt de obicei folosite, deoarece sunt destul de eficiente din punct de vedere energetic și pot fi foarte mici și ușoare. Prințipiu după care funcționează sistemele de vizionare stereo sunt destul de simplu și se bazează pe găsirea punctelor corespunzătoare între imaginile camerei din stânga și cea dreaptă și folosind distanța dintre ele (disparitate binoculară) pentru a deduce adâncimea punctului. Procesul de potrivire stereo (*stereo matching*) este un compromis dintre complexitatea computațională și densitatea hărții de adâncime. Una din probleme este că, în regiunile cu textură scăzută sunt greu de perecheat, din cauza lipsei de caracteristici între cele două imagini stereo, pentru a găsi corespondența lor.

Există și de asemenea, limite fizice la acurateția unui sistem stereo. Algoritmii de potrivire nu reușesc în regiunile ocluse, vizibilă la una dintre camere, dar nu și de celălaltă. Nu există posibilitatea de a găsi corespondențe, având în vedere că regiunea este ascunsă într-una din imaginile stereo. În plus, distanța maximă măsurabilă de către sistemul stereo este invers proporțional cu distanța dintre cele două obiective ale camerelor. Din aceaste probleme, un sistem de vedere stereo are o limită maximă, și o limită minimă în același timp. Si potrivirea stereo devine imposibilă la distanțe mici, din cauza ocluziei excesive, deoarece obiectele încep a

pare diverit în imaginile stereo, la fel la distanță maximă diferența dintre cele două imagine devine de nerecunoscut.

Unele dintre limitările importante ale viziunii stereo ar putea fi depășite prin completarea acesteia cu estimarea adâncimii monoculare. Estimarea adâncimii monoculare se bazează pe exploatarea atât a proprietăților locale ale texturii, a gradientelor și a culorilor, cât și a relațiilor geometrice globale, cum ar fi plasarea obiectului relativ și indicii de perspectivă, fără constrângerile a priori asupra intervalelor minime și maxime, ocluziile. Este argumentat că estimarea adâncimii monoculare poate fi utilizată pentru a îmbunătăți un algoritm de vizionare stereo: prin utilizarea informațiilor complementare, acesta ar trebui să furnizeze estimări de profunzime mai precise în regiunile de ocluzie și de potrivire stereo de încredere scăzută.

1.2 Data seturi de adâncime

O parte esențială în învățarea automată este datele utilizate pentru antrenare și testare. Diverse seturi de date RGBD au devenit standard în literatură, fiind folosite pentru măsurarea performanțelor algoritmilor. Mergem la lucrările recente ale lui Firman și alții [22], pentru un studiu cuprinzător al seturilor de date existente și a predicțiilor pe viitor pentru viitor.

Saxena et al. [3, 2] au publicat setul de date pe care l-au folosit pentru a-și forma algoritmul Make3D, constând din perechi de imagini în aer liber și date de adâncime, colectate utilizând un scanner 3D cu laser. Imaginele sunt $2,272 \times 1,704$ în rezoluție, în timp ce hărțile adâncime sunt de 55×305 . Este împărțită canonice într-un eșantion de antrenare de 400 de imagini și 134 de testare.

Silberman et al. [23] a introdus setul de date NYU Depth Dataset V2, format din 1449 imagini RGBD cu ajutorul unui senzor Kinect într-o gamă largă de scene de interior. În plus de hărți de adâncime, setul de date este puternic anotat cu etichete per-pixel obiect, precum și tipul de suport. Atât imaginile cât și hărțile de adâncime au rezoluția de 640×480 pixeli.

În 2012, Geiger et al. [16] a dezvoltat suita de referință pentru viziunea KITTI, destinată evaluării multiple sarcini de computer vision, și anume algoritmi de potrivire stereo și flux optic și sarcini la nivel înalt cum ar fi odometria vizuală, SLAM și detectarea și urmărirea obiectelor. Setul de date este format din perechi de imagini stereo în aer liber și hărți de adânci de însoțire, obținute cu ajutorul unui scanner laser pe partea superioară a unei mașini în mișcare. În 2015, Menze et al. [24, 25] a îmbunătățit la KITTI cu un flux suplimentar stereo, optic și set de date pentru fluxul de scenă, colectate folosind mijloace similare, dar care prezintă scene dinamice, nu statice. Setul de date este împărțit în 200 imagini de antrenamente și 200 de testare, pentru care nu există hărți de adâncime adevărate făcute publice, iar imaginile și hărțile de adâncime sunt de aproximativ 1242×375 în rezoluție, cu

variații mici între cadre datorită factorilor diferenți de crop în timpul calibrării și a corectării de distorsionare.

Mai recent, Mayer et al. [26] au lucrat pe trei seturi de date sintetice sterile masive, totalizând peste 35.000 de perechi de imagini, care, la fel ca KITTI, au scopul de a furniza discrepanță reală, flux optic și hărți de scenă de flux, făcându-le potrivite pentru o gamă largă de aplicații în computer vision. Cele trei seturi de date constau în scene animate ale obiectelor de zbor random, un scurtmetraj animat și o mașină de condus KITTI. Numărul mare și variabilitatea eșantioanelor din setul de date îl fac adecvat pentru antrenarea unei arhitecturi de rețele neuronale profunde, pe care autorii le demonstrează prin obținerea rezultatelor de ultimă generație în domeniul stereo, estimarea disparităților și rezultate promițătoare în estimarea fluxului de scenă.

Chen et al. [37] a făcut crowd-source la o mulțime de date de adnotări relative de adâncime, utilizând imagini din Flickr cu setări nestructurate. Persoanele care au participat în croud sourceing au fost prezentatăi o imagine și două puncte evidențiate și au fost întrebați care dintre cele două puncte a fost mai aproape. Punctele au fost fie eșantionate radnom din întreaga imagine, fie de-a lungul aceleiași linii orizontale și simetrice în raport cu centrul, și au fost obținute în total mai mult de 500 000 de perechi. Acest set de date este util pentru metodele care se bazează pe măsurători de adâncime relativă, mai degrabă decât absolută.

2 Contribuții

3 Metodologii

În acest capitol, oferim o privire în mai amănunte asupra celor mai importante contribuții din literatură cu privire la estimarea adâncimii intr-o imagine.

3.1 Random Markov Field

În Saxena et. al [1] se ia abordarea de învățare supervizată a acestei probleme, în care încep prin colectarea unui set de imagini monoculare de antrenament cu harta lor de adâncime. Apoi, se aplică învățarea supervizată pentru a prezice harta de adâncime ca o funcție a imaginii. Modelul lui Saxena utilizează un Markov Random Field (MRF) antrenat în mod discriminativ, care încorporează caracteristici multiscale locale și globale și modelează atât adâncimi la puncte individuale, cât și relația dintre adâncimi în diferite puncte. Arată că, chiar și pe scene nestructurate, algoritmul dat este în măsură să recupereze adâncime destul de precisă.

Au început cu folosirea unui scanner de distanță 3-D pentru a colecta date de antrenament, care conțin un set mare de imagini și adâncimea lor corespunzătoare. Folosind acest set de antrenament, MRF este invățat în mod supervizat pentru a prezice profunzimea; astfel, mai degrabă decât modelarea distribuției comune a caracteristicilor și adâncimilor imaginii, modelează doar distribuția posterioară a adâncimilor date ale caracteristicile imaginilor. Modelul de bază folosește termenii L2 (Gaussian) în potențialul interacțiunii MRF și captează adâncimile și interacțiunile între adâncimi la dimensiuni spațiale multiple. Mai este prezentat un al doilea model care utilizează potențialul de interacțiune L1 (Laplacian). Învățarea în acest model este aproximativă, dar inferența posterioară MAP exactă este tractabilă (similară MRF-urilor Gaussian) prin programare liniară și oferă o mai mare adâncime decât modelul simplu Gaussian.

3.1.0.1 Rezultate

Algoritmul a fost invățat și testat pe un set de imagini cuprinzând din mai multe medii(copaci, tufișuri, clădiri, oameni și copaci, locuri de interior etc.). Tabelul Table 3.1 prezintă rezultatele testului atunci când se utilizează combinații de caracteristici diferite. Vedem că utilizarea funcțiilor *multiscale* și a coloanelor îmbunătățește semnificativ performanța algoritmului. Cu includerea termenilor de interacțiune și-a îmbunătățit performanțele, iar modelul Laplacian funcționează mai bine decât cel

Features	Toate	Padure	Campus	Interior
Baseline	0.296	0.283	0.343	0.228
Gaussian (S1,S2,S3,H1,H2,fara vecini)	0.162	0.159	0.166	0.165
Gaussian (S1, H1, H2)	0.171	0.164	0.189	0.173
GAUSSIAN (S1,S2, H1,H2)	0.155	0.151	0.164	0.157
GAUSSIAN (S1, S2,S3, H1,H2)	0.144	0.144	0.143	0.144
GAUSSIAN (S1,S2,S3, C, H1)	0.139	0.140	0.141	0.122
GAUSSIAN (S1,S2,S3, C, H1,H2)	0.133	0.135	0.132	0.124
LAPLACIAN	0.132	0.133	0.142	0.084

Table 3.1: Efectul caracteristicilor multiscale și caracteristicile coloanelor asupra preciziei. Eroarea medie absolută (Erorile RMS au dat rezultate similare) sunt pe o scară logaritmica (baza 10). H1 și H2 reprezintă statistici sumare pentru $k = 1, 2$. S1, S2 și S3 reprezintă cele trei scale. C reprezintă funcțiile coloanei. Nivelul de bază este invatat numai cu termenul de părtinire (fără caracteristici).

Gaussian. Și modelul Laplacian dă hărți de adâncime cu granițe semnificativ mai clare. Tabelul Table 3.1 prezintă erorile obținute de algoritmul dat pe o varietate de imagini de pădure, campus și interior. Rezultatele arată că algoritmul estimează mapările de adâncime cu o eroare medie de 0.132 ordine de mărime. De asemenea, pare a fi foarte robust față de diferența dintre locurile umbrite. Algoritmul prezice destul de bine adâncimile relative ale obiectelor. Dar este limitat pentru adâncimea absolută. Erorile pot fi atribuite la limita de precizie a setului de antrenament, și la obiectele iregulare din imagine ca exemplu frunzele copacilor.

3.2 Abordare non-parametrică

Karsch et al. [6] au prezentat un framework pentru extragerea hărților de adâncime dintr-o singură imagine și, de asemenea, o adâncime temporară consecventă din secvențe video, robuste pentru mișcarea camerei, modificări ale distanței focale și peisaj dinamic. Abordarea lor este non-parametrică, bazată pe transferul de adâncime de la imaginile de intrare similară într-o bază de date RGBD existentă, prin potrivirea și deformarea celei mai apropiate hărți de adâncime a candidatului, apoi interpolarea și netezirea hărții de adâncime printr-o procedură de optimizare pentru a garanta consistența spațială. Pentru secvențele video, se adaugă termeni suplimentari la funcția de cost care penalizează inconsecvențele temporale. Algoritmul depinde de baza de date RGBD care este disponibilă la timpul de execuție și necesită astfel cantități mari de memorie.

Pe scurt utilizează un mecanism de transfer kNN bazat pe fluxul SIFT [39] pentru a estima adâncimile fundalurilor statice din imaginile singulare, pe care le mărește cu informații de mișcare pentru a estima mai bine subiectele din prim-plan în video-

clipuri. Acest lucru poate duce la o aliniere mai bună, însă necesită ca întregul set de date să fie disponibil la runtime și să efectueze proceduri costisitoare de aliniere.

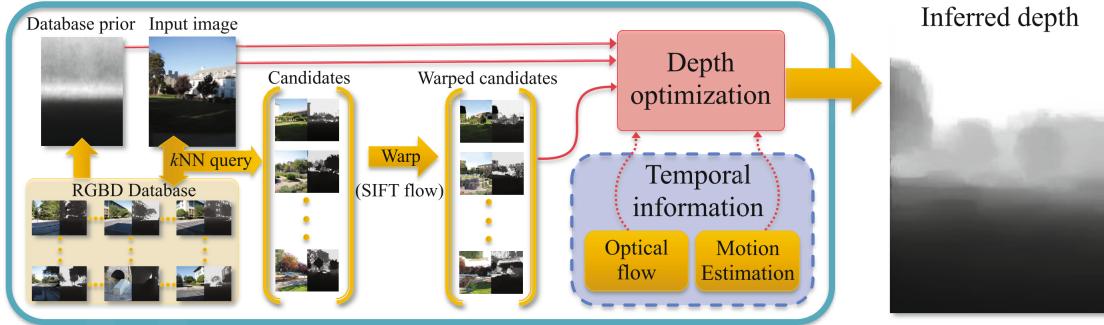


Figure 3.1: Pipeline-nul de estimare a adâncimei

Se folosește lucrările recente de învățare nonparametrică [27], care evită de a defini explicit un model parametric și necesită asumții. Această abordare se scalează, mai bine în ceea ce privește dimensiunea datelor de antrenament, fără a necesita practic timp de antrenament. Tehnica dată dându-i problema extractiei adâncimei video, nu impune nici o cerință asupra videoclipului, cum ar fi parallaxul motivației sau lungimea secvenței, și poate fi aplicată unei singure imagini.

Pipeloul dat în Figure 3.1 pentru estimarea adâncimii. Având o imagine de intrare, se găsesc candidați, care se potrivesc cu imaginea de input în baza de date și candidații sunt *warp-upiți* să se potrivească cu structura imaginii de intrare. Apoi se folosește o procedură globală de optimizare pentru a interpola candidații warp-ati, producând estimări de adâncime pe pixeli pentru imaginea de intrare. Cu ajutorul informațiilor temporale (de exemplu, extrase dintr-un videoclip), algoritmul dat poate obține o adâncime temporală coerentă mai exactă.

Rezultate Metoda dată la vremea ei era la fel de bună sau mai bună decât cea de ultimă generație pentru fiecare metrică. Adâncimea estimată este suficient de bună pentru a genera imagini 3D convingătoare. În unele cazuri, metoda dată produce chiar mai multe estimări de adâncime decât adâncimea adevărată (cu erori de rezoluție scăzută și senzori). Structurile subțiri (de exemplu, arbori și stâlpi) sunt de obicei recuperate bine; totuși, structurile fine sunt ocazional ratate datorită regularizării spațiale.

3.3 CNN cu două componente, *coarse(aspru)* global și *fine(fin)* local

În Eigen et al. [9] este prezentată o nouă abordare pentru estimarea adâncimii dintr-o singură imagine. Adâncimea este regresat în mod direct folosind o rețea neurală cu două componente: una care evaluează mai întâi structura globală a scenei, apoi după care o rafinează folosind informațiile locale. Rețeaua este învățată folosind o pierdere care explică explicit relațiile de adâncime dintre locațiile pixelilor, în plus eroarea point-wise.

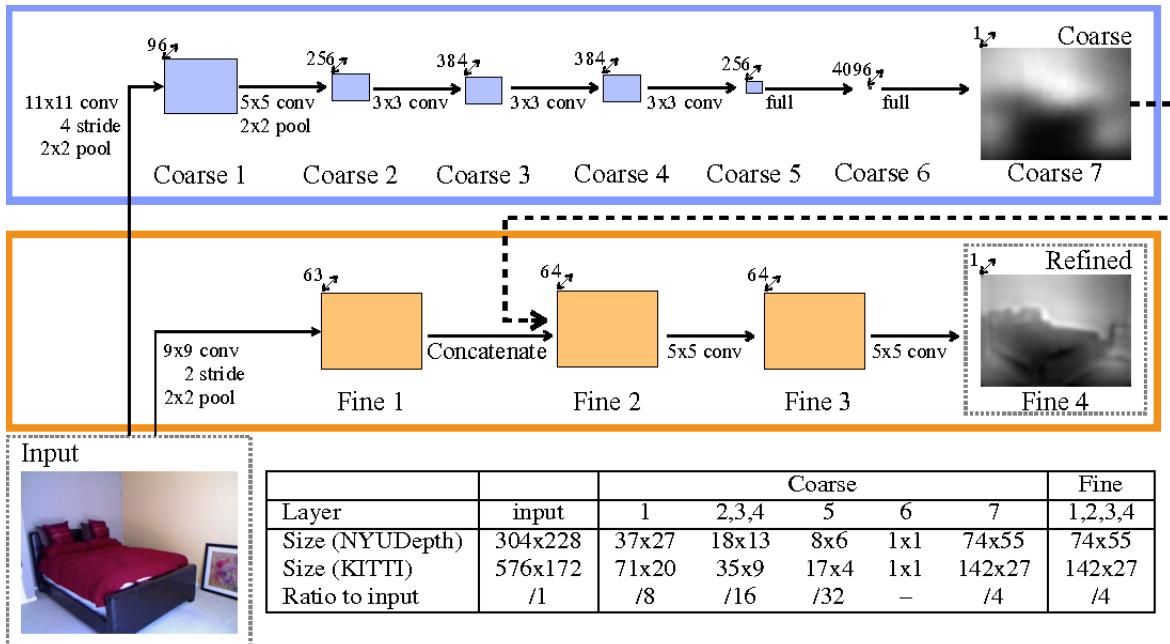


Figure 3.2: Modelul arhitecturii

Prezentată în (Figure 3.2) este rețeaua la scară (*aspru*)*coarse* ce prezice profunzimea scenei la nivel global. Acest lucru este apoi rafinat în regiunile locale printr-o rețea la scară (*fină*)*fine*. Ambele stive sunt aplicate la intrarea originală, dar în plus, ieșirea rețelei *coarse* este trecută la rețeaua fină ca și caracteristici suplimentare de imagine de prim strat. În acest fel, rețeaua locală poate edita predicția globală pentru a include detalierea obiectelor.

Sarcina rețelei *coarse-scale* este de a prezice structura generală a hărții de adâncime utilizând o vedere globală a scenei. Straturile superioare ale acestei rețele sunt complet conectate și, astfel, conțin întreaga imagine în câmpul vizual. În mod similar, straturile inferioare și de mijloc sunt concepute să combine informații din diferite părți ale imaginii prin operațiuni de *max-pooling* la o dimensiune spațială mică. Realizând acest lucru, rețeaua este capabilă să integreze o înțelegere globală

a întregii scene pentru a afla adâncime. O astfel de înțelegere este necesară în cazul imaginii unice pentru a utiliza în mod eficient indicii, cum ar fi punctele de dispariție, locațiile obiectelor și alinierea camerei. O vizualizare locală (așa cum este utilizat pentru stereo matching) este insuficientă pentru a observa caracteristici importante precum acestea.

După ce este luat o perspectivă globală pentru a prezice harta de adâncime coarse, se realizează rafinări locale utilizând o rețea fine-scale. Sarcina acestei componente este de a edita predicția *coarse* pe care o primește, pentru a alinia detaliile locale, cum ar fi marginile obiectului și pereților. Stacurile ale rețelei *fine-scale* constă numai din straturi convoluționale, împreună cu o etapă de pooling pentru caracteristicile de margine ale primului strat. Mai concret, ieșirea grosieră este alimentată ca o hartă a caracteristicilor suplimentare la nivel scăzut.

Rezultate La antrenament cu data setul NYU Depth [23] cu un set de 120 mii de imagini, care sunt amestecate într-o listă de 220 mii după egalarea distribuției scenei (1200 pe scenă). A fost testat pe 694 set de imagini de testare din NYU. Cele mai bune rezultate de vreme a antrenamentului a fost realizat în 38 ore pentru rețeaua coarse și 26h pentru cea fine, pentru un total de 2,6 zile utilizând un NVidia GTX Titan Black. Predicția testului durează 0,33s pe lot (0,01s / imagine).

Setul de antrenament din KITTI [16] are 800 de imagini pe scenă. Excluzând fotografile în cazul în care mașina este staționară (accelerația mașinii sub un prag) pentru a evita dublarea. Sunt utilizate camerele RGB de stânga și dreapta, dar sunt tratate ca fotografii neasociate. Setul de antrenament are imagini unice de 20 mii, pe care le amestecăm într-o listă de 40 mii (inclusiv duplicate) după seară egalarea distribuției scenei. Antrenamentul a durat 30 de ore pentru modelul coars și 14 ore pentru ce fine; testarea predicției durează 0,40s / lot (0,013s / imagine).

Ceea ce este interesant de observat este că rețeaua "*coarse*" + "*fine*" este un exemplu mai amplu de rețea reziduală care a fost folosită în lucrările ulterioare (Laina et al.)[8]. Se pare că o bună modalitate de a obține rezultate bune este să concatenăm hărțile caracteristicilor la un nivel superior cu o estimare "*coarse*", care permite un sentiment de partajare a caracteristicilor între rețele. În lucrările ulterioare [4], ele dezvoltă o rețea mai generală, cu trei scale de rafinament, care se aplică apoi sarcinilor de estimare a adâncimii, estimarea normalelor a suprafeței și etichetării semantice.

3.4 Problemă de optimizare discretă-continuă

Liu et al. [7] formulează estimarea adâncimii monoculare ca problemă de optimizare discret-continuă, unde variabilele continue codifică adâncimea superpixelilor în imaginea de intrare, iar cele discrete reprezintă relațiile dintre superpixelurile vecine. Soluția la această problemă de optimizare continuă discret este obținută

Metoda		rel	\log_{10}	rms
Karsch et al. [6]	C_1	0.355	0.127	9.2
	C_2	0.361	0.148	15.1
Liu et al. [7]	C_1	0.335	0.137	9.49
	C_2	0.338	0.134	12.6

Table 3.2: Reconstruirea erorii adâncimii pe baza de date Make3D pentru Karsch et al. [6] și pentru metoda Liu et al. [7] pe două criterii C_1 și C_2

prin efectuarea inferenței într-un model grafic folosind propagarea de convingere (*belief propagation*). Potențialele unare din acest model grafic sunt calculate prin utilizarea imaginilor cu adâncime cunoscută.

Se utilizează o abordare nonparametrică pentru a recupera hărți adânci candidate. În special, se recuperă imaginile K cele mai apropiate de imaginea de intrare dintr-un set de imagini pentru care este cunoscută adâncimea. În acest scop, se efectuează o căutare cu vecinul cel mai apropiat bazată pe funcțiile GIST, PHOG și Object Bank concatenate și folosesc direct profunzimea imaginilor preluate. Hărțile de adâncime K recuperate apoi acționează direct ca stări în prima rundă de PCBP, adică în această rundă nu sunt utilizate probe random. Mai important, se introduce utilizarea unor variabile discrete care permit să fie modelat relații mai complexe între superpixelii vecini și să formuleze estimarea adâncimii ca inferență într-un model grafic discret-continuu. După cum reiese din rezultate, această formulă este benefică în ceea ce privește precizia adâncimii estimate și sa dovedit a fi eficientă atât pentru scenariile de interior, cât și pentru cele în aer liber.

În tabelul Table 3.2 arată erorile caclulării adâncimii a Karsch et al. [6] care în 2012 era în frunte, ca o metodă de ultimă generație. După aceste numere metoda dată a avansat cu câteva procente, 5.6% a erorii rel. (C_1) semnifică erori sunt computate în regiunile cu adâncime adevărat mai mică de 70 m, iar (C_2) semnifică că erorile sunt calculate în întreaga imagine.

3.5 Regresie la caracteristicile adânci, și CRF-uri ierarhici

În Li et al. 2015 [34] se abordează problema adâncimii prin regresia trăsăturilor rețea neuronală profund convolutivă (DCNN), prin combinarea cu un pas de rafinare la post-procesare folosind un câmp random condițional (CFR). Framework-ul dat lucrează la două nivele, superpixel și nivelul pixel. În primul rind, este construit un model DCNN pentru a invăța maparea de la *patch-urile multi-scale* a imaginii pentru a estimarea adâncimei ori valorile normelor de suprafață la nivel de superpixel. În al doilea rind, adâncimea de super-pixel estimată sau normala suprafeței este îmubătățită la nivel de pixel prin exploatarea diverselor potențiale pe hărțile de

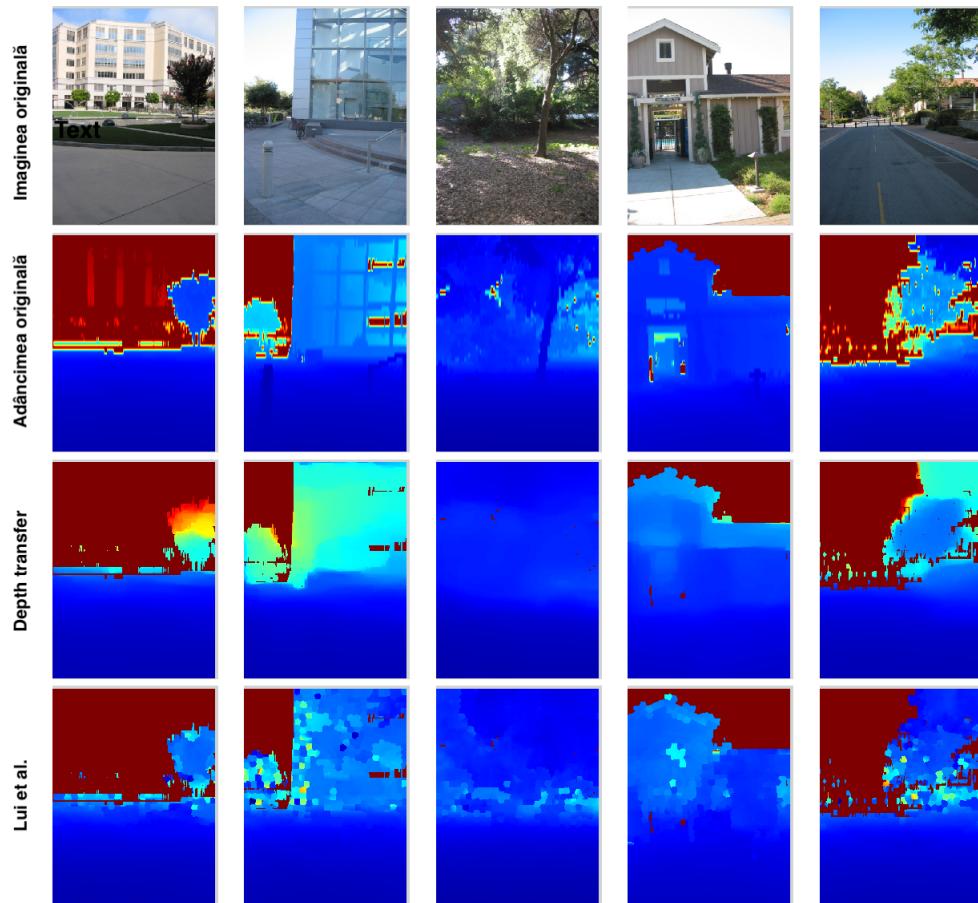


Figure 3.3: Comparația calitativă a adâncimilor estimate cu depth transfer (Karsch et al.) [6] și cu metoda dată (Liu et al.) [7] din setul de date Make3D. Culoarea indică adâncimea (roșu este departe, albastru este aproape)

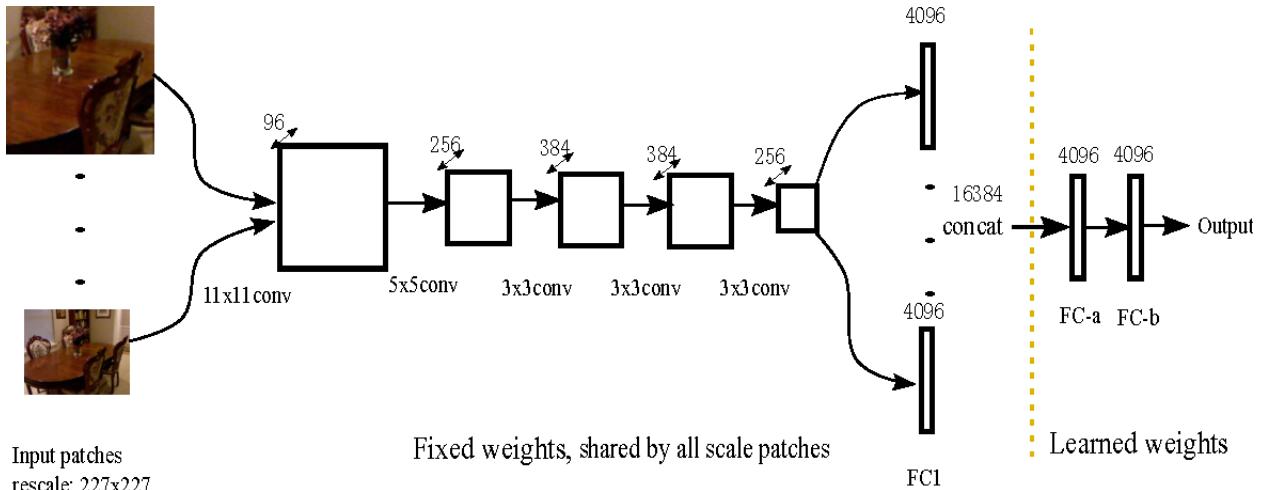


Figure 3.4: Vizualizarea framework-ului multiscale

adâncime ori normala suprafețelor, care include un termen de date, un termen de netezime între super-pixeli și un termen de auto-regresie, caracterizând structura locală a hărții de estimare.

Abordarea la estimarea adâncimii imaginii la nivel de pixel constă în două etape: regresia adâncimii la super-pixeli și rafinarea adâncimii de la super-pixeli la pixeli. În primul rând, se formulează o estimare a adâncimii de nivel super-pixel ca o problemă de regresie. Având o imagine, se obține super-pixeli. Pentru fiecare super-pixel, se extrage *patch-uri multi-scale* în jurul centrului super-pixel. Un CNN adânc este apoi învățat să codifice relația dintre patch-urile de input și adâncimea corespunzătoare. În al doilea rând, se rafinează estimarea adâncimii de la nivelul super-pixel la nivelul pixelilor prin inferență pe un contur random condițional ierarhic (CRF). Diferite potențiale sunt luate în considerare atât la nivele de super-pixeli, cât și la nivele de pixeli. Este important faptul că problema inferenței MAP are o soluție în formă închisă. O arhitectură generală CNN este prezentată în figura Figure 3.4. Partea cu greutăți fixe a CNN-ului poate fi transferată de la modele pre-instruite ale lui Krizhevsky et al. cum ar fi AlexNet [35] sau VGGNet mai profund [36].

Rezultate

Pentru antrenament se folosesc bazele de date Make3D [3, 2] și NYU [23], se utilizează SLIC pentru a extrage super-pixelii. În legătură cu eroarea medie relativă, abordarea dată a îmbunătățit-o cu 33.4 % față de Liu et al. [7], dar în comparație cu Eigen et al. [9] diferența de eroare nu este semnificativă.

3.6 Învățare reziduală

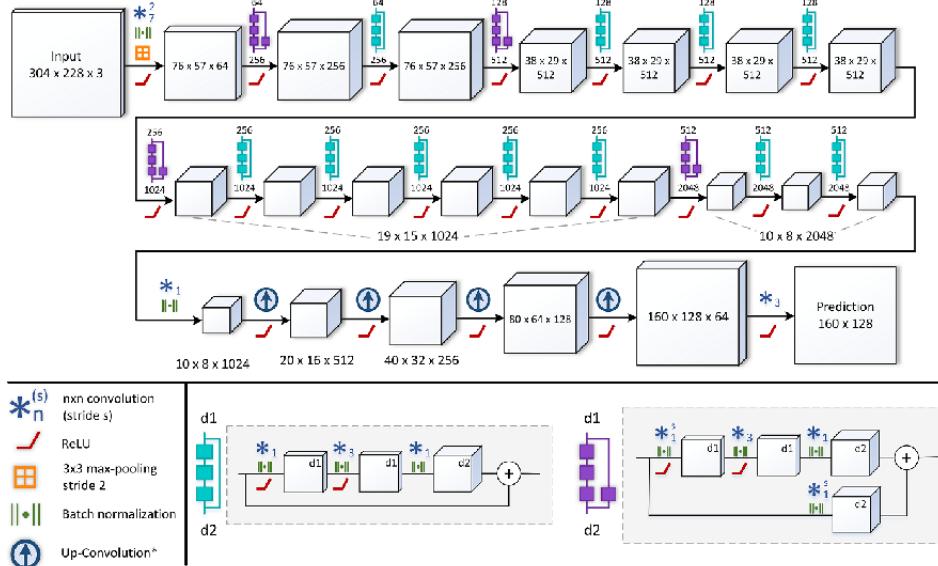


Figure 3.5: Arhitectura rețelei

3.6 Învățare reziduală

În Laina et al. [8] este propusă o arhitectură complet conlovuțională, care cuprinde învățarea reziduală, pentru a modela adâncimea ambiguă între imaginile monoculare și hărțile de adâncime. Pentru a îmbunătăți rezoluția de ieșire, este prezentată o modalitate nouă de a învăța eficient harta de caracteristici de *up-sampling* în rețea. Pentru optimizare, se introduce pierderea Huber inversă care se potrivește foarte bine la metoda dată și este condusă de distribuțiile a valorilor prezente în hărțile de adâncime. Modelul dat este compus dintr-o singură arhitectură care este învățată de la un capăt la altul și nu se bazează pe tehnici de post-procesare, cum ar fi CRF sau alte etape de rafinare suplimentare. Ca rezultat, rulează în timp real pe imagini sau videoclipuri.

Mai întâi, se introduce o arhitectură complet conlovuțională pentru predicția profunzimii, dotată cu noi blocuri de *up-sampling*, care permit deducerea hărților de adâncime cu o rezoluție densă, în același timp, necesită mai puține parametri și folosește la antrenament o ordine de mărime mai puține date decât metodele propuse precedent. Schema prousă este mai eficientă pentru *up-convolution* și este combinată cu conceptul de învățare reziduală [31], pentru a crea blocuri up-projection pentru un upsampling mai efectiv a hărților de caracteristici. *Up-convolution* se referează la blocurile ce angajează straturile *unpooling* care performă operația reversă de *pooling*, implementarea straturilor de *unpooling* este bazată pe [32]. În cele din urmă, rețeaua este antrenată optimizând pierderea bazat pe funcția Huber reversă (berHu) [30] ce demonstrează, teoretic și experimental că este benefică și potrivită pentru predicția adâncimii.

Arhitectura propusă poate fi văzută în figura Figure 3.5. Ca exemplu, în această imagine, dimensiunile hărților de caracteristici corespund rețelei antrenate pentru input cu mărimea 304×228 , în cazul setului de date NYU Depth v2 [23]. Prima parte a rețelei se bazează pe ResNet-50 și este inițializată cu greutăți pre-antrenate. A doua parte a arhitecturii date ghidează rețeaua în învățarea *upsampling* printr-o succesiune de straturi *unpooling* și conveționale. Urmărind seturile acestor blocuri *upsampling*, *dropout-ul* este aplicat și urmat de un strat final convețional obținându-se predictia hărții de adâncime.

Resultate Pentru implementarea rețelei date, se folosește MatConvNet [28] și este antrenată pe un singur NVIDIA GeForce GTX TITAN cu memorie GPU de 12 GB. Straturile de greutate ale părții *down-sampling* a arhitecturii sunt inițializate de modelele corespunzătoare ResNet [29]. Straturile adăugate recent ale părții *upsampling* sunt inițializate ca filtre random prelevate dintr-o distribuție normală cu zero și 0.01 variație. Modelul este antrenat pe ambele base de date NYU Depth Dataset[23] și Make3D [3, 2]. Pe baza de date NYU, în comparație Eigen et al. [9] eroarea relativă este îmbunătățită cu 40%, Liu et al [33] cu 60 %, Eigen și Fergus [4] cu 20 %. Pe baza de date Make3D [3, 2] în comparație cu Karsch et al. [6] eroarea relativă este îmbunătățită cu 50.4%, Li et al. [34] cu 36.6 %.

3.7 Chen et al. [37] 2016

În Chen et al. [37] problema adâncimii dintr-o imagine, este redefinită ca recuperarea profunzimii dintr-o singură imagine realizată în setări fără restricții. Autorii introduc un nou set de date numit "Depth in the Wild" constând în imagini în sălbăticie adnotate cu adâncimea relativă între perechi de puncte random. Propun și un algoritm nou ce învăță a estima adâncimea folosind anotațiile adâncimilor relative.

Construcția bazei de date se face prin adunarea imaginii de pe Flickr. Utilizăm cuvinte cheie random prelevate dintr-un dicționar englez și se exclude imagini artificiale, cum ar fi desene și clipuri artistice. Pentru a aduna adnotări de profunzime relativă, este prezentat un lucrător, o imagine, și două puncte evidențiat, și se întrebă "care punct este mai aproape, punctul 1, punctul 2 sau greu de spus?" Lucrătorul apasă un buton pentru a răspunde.

Estimarea adâncimii este luată o abordare mai simplă. Ideea este că orice algoritm de le imagine la adâncime ar trebui să computeze o funcție care să mapeze o imagine, la o adâncime *pixel-wise*. Această funcție poate fi reprezentată ca o rețea neuronală care să fie învățată de la capăt la catăt. Pentru acest lucru, este nevoie doar de două componente: (1) un design de rețea care produce aceeași rezoluție ca intrarea și (2) o modalitate de a instrui rețeaua cu adnotări de adâncime relativă.

Rețelele care generează aceeași rezoluție ca și intrarea sunt excelente, inclusiv modelul recent de estimare a adâncimii [8, 35] și cele pentru segmentarea semantică și de-

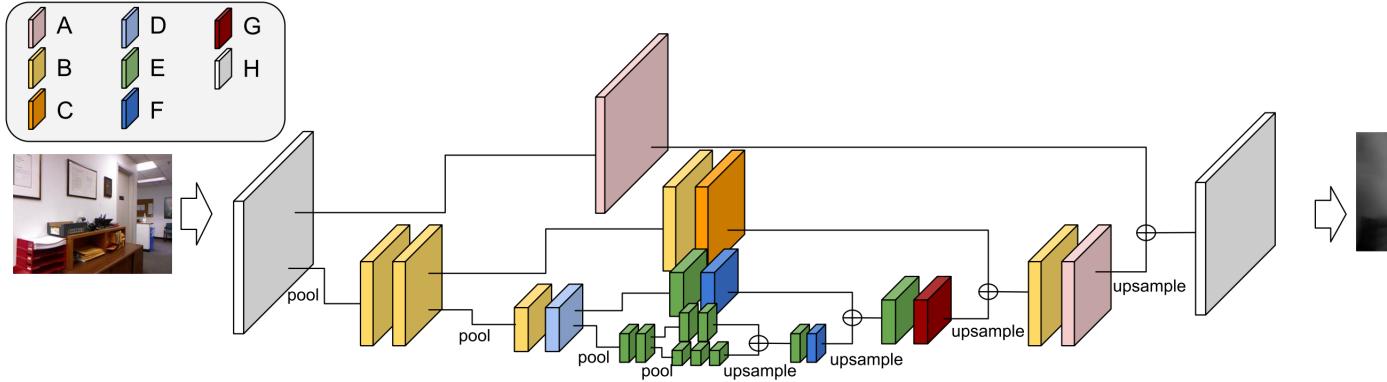


Figure 3.6: Design de rețea. Fiecare bloc reprezintă un strat. Blocurile care împărțășesc aceeași culoare sunt identice. Semnul \oplus indică adăugarea elementului. Blocul H este o conoluție cu filtru 3x3

tectarea marginilor. Un element comun este prelucrarea și transmiterea informațiilor în mai multe scale.

În cadrul acestei lucrări, se utilizează o variantă a rețelei de "hourglass" introdusă recent Figure 3.6, care a fost utilizată pentru a obține rezultate de ultimă generație în estimarea poziției uamne [38]. Se compune dintr-o serie de conoluții și de *downsampling*, urmată de o serie de conoluții și de *upsampling*, intercalate cu conexiuni de *skip* care adaugă caracteristici de la rezoluții înalte. Forma simetrică a rețelei seamănă cu o clepsidră, de unde și numele.

Inovația în această lucrare constă în combinarea unei rețele adânci care face o predicție *pixel-wise* și o *ranking loss* plasată pe predicția *pixel-wise*. O rețea profundă care face o predicție de tip *pixel-wise* nu este nouă și nici o *ranking loss*. Dar, din câte știu, o astfel de combinație nu a fost propusă înainte și, în special, nu pentru estimarea adâncimii. Este arătat că algoritmul dat depășește performanțele predecesorilor, iar algoritmul dat, combinat cu datele existente RGB-D și noile adnotări relative de adâncime, îmbunătățește semnificativ percepția de adâncime a imaginii unice în sălbăticie. Comparativ cu metodele precedente, algoritmul dat este mai simplu și are performanțe mai bune.

3.8 Învățarea Nesupervizată

3.8.1 Autoencoder

Garg et al. [40] propune un framework nesupervizat pentru a învăța o rețea neuronală profund convolutivă pentru predicția adâncimii într-o singură imagine, fără a necesita o etapă de pre-pregătire sau hărți de adâncime anotate adevărate. Ci

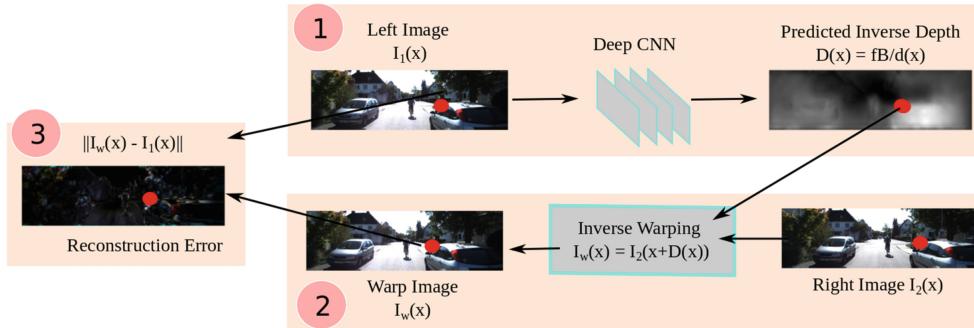


Figure 3.7: Arhitectura metodei

realizează acest lucru prin formarea rețelei într-o manieră analoagă cu un autoencoder. La timpul de antrenament, să considerăm o pereche de imagini, sursă și țintă, cum ar fi o pereche stereo. Își encoderul convolutional este antrenat pentru sarcina de a prezice adâncimea din imaginea sursă.

Pentru a face acest lucru, se generează în mod explicit un *warp* invers a imaginii țintă utilizând adâncimea estimată și deplasarea cunoscută a inter-vizualizării, pentru a reconstrui imaginea sursă; eroarea fotometrică în reconstrucție este pierderea de reconstrucție a encoderului. Achiziția acestor seturi de antrenament este considerabil mai simplu în comparație cu modelele similare.

Pierderea este linearizată folosind aproximarea Taylor de primă ordine și necesită, prin urmare, antrenare coarse-la-fine.

3.8.2 Reconstucție a imaginii în timpul antrenamentului

În Godard et al. [41], este luată abordare alternativă și se tratează estimarea automată a adâncimii ca problemă de reconstrucție a imaginii în timpul antrenamentului. Modelul dat complet convolutional nu necesită date de adâncime și este instruit să sintetizeze adâncimea ca intermediar. Învață să prezică corespondența la nivel de pixel între perechi de imagini stereo rectificate, care au o linie de bază a camerei cunoscute. Mai prezintă și o verificare a consistenței stânga-dreapta încorporată, care permite modelul să fie invățat pe perechi de imagini fără a fi nevoie de supraveghere sub forma de hărți de adâncime avevărate.

Dând o singura imagine în timp de testare, scopul este de a învăța o funcție să prezică adâncimea scenei per pixel. Intuiția aici este ca având o pereche de camere binoculare calibrate, această funcție trebuie să poată reconstrui o imagine din alta. Atunci, se poate de presupus că a fost invățat ceva despre forma 3D a scenei din aceste imagini. În loc de a preciza adâncimea direct, în timpul de antrenament, se încearcă de a găsi cîmpul dens de corespondență, care poate fi aplicat la imaginea stîngă de a reconstrui imaginea dreapă. Respectiv se poate de a estima imaginea stîngă având imaginea

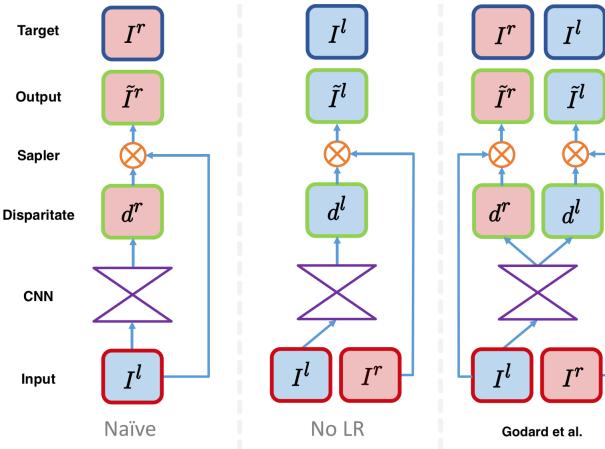


Figure 3.8: Strategii de *sampling* pentru maparea inversă

drapă. Disparitatea dintre aceste două imagini poate fi transformată într-o hartă de adâncime.

La un nivel înalt, rețeaua dată estimează profunzimea prin deducerea disparității care deviază imaginea din stânga pentru a se potrivi cu cea din dreapta. Esența principală este că se poate deduce simultan ambele disparități (de la stânga la dreapta și de la dreapta la stânga), folosind doar imaginea de intrare din stânga, și obținerea unor adâncimi mai bune prin impunerea acestora pentru a fi consistentă una cu cealaltă.

Rețeaua dată generează imaginea previzionată cu mapare inversă utilizând un *sampler* bielinar, rezultând un model complet diferențiabil de formare a imaginii. După cum este ilustrat în Figure 3.8, învățarea naivă de generare a imaginei drepte din partea stângă prin *sampling*, va produce disparități aliniate cu imaginea dreaptă. Cu toate acestea, se dorește ca harta de disparități output să se alinieze cu imaginea de input stânga, adică rețeaua trebuie să fie probată din imaginea corectă. Mai este posibilitatea de a învăța rețeaua pentru a genera vederea din stânga prin eșantionare din imaginea dreaptă, creând astfel o hartă de disparitate aliniată la vederea stângă (no LR în Figure 3.8). Dar, disparitatea dedusă manifestă artefakte de 'copia texturii' și erori la adâncime. Acest lucru se poate rezolva prin formarea rețelei de a prezice hărțile de disparitate, pentru ambele vederi, prin *sampling-nul* din imaginile de intrare opuse. Acest lucru necesită doar o singură imagine stângă ca intrare în straturile conoluționale, iar imaginea corectă este utilizată numai în timpul antrenamentului (metoda Godard în Figure 3.8). Impunerea consistenței între ambele hărți de disparitate utilizând acest nou cost consistent stânga-dreapta, conduce la obținerea unor rezultate precise.

Această arhitectură total conoluțională este inspirată de Disp-Net [39], dar prezintă câteva modificări importante care permit învățarea fără a cere adâncime ade-

vărată. Rețeaua este compusă din două părți principale - un *encoder* și un *decoder*. Decodorul folosește conexiuni *skip* [47] din blocurile de activare ale *encoder-ului*, permitându-i să deducă detalii de rezoluție mai ridicată. La output se dă predicția disparității la patru scări diferite, care dublează rezoluția spațială la fiecare dintre scările ulterioare. Chiar dacă este nevoie doar de o singură imagine, rețeaua dată prezice două hărți de disparități la fiecare scală de output - de la stânga la dreapta și de la dreapta la stânga.

Rezultate

Această metodă a fost antrenată și testată pe mai multe tipuri de date-seturi. ca KITTI [16] și Cityscape[17]. În comparație cu metodele Eigen et al.[9] și Liu et al.[7] antrenate și testate pe setul de date KITTI [16], metoda dată îi depășeste cu o îmbunătățire a erorii de aprox. 10 %. A avut succes și la testare pe setul de date de la Make3D, în ciuda faptului că antrenarea a fost făcută pe Cityscape, cu rezultate rezonabile.

Chiar dacă verificarea consistența stînga-dreapta, și post procesarea îmbunătățește calitatea rezultatelor, există încă câteva obiecte vizibile la limitele de ocluzie, deoarece pixelii din regiunea de ocluzie nu sunt vizibili în ambele imagini. și metoda dată necesită perechi stereo rectificate și aliniate temporal în timpul antrenamentului, ce înseamnă că nu e posibil de folosit seturi de date de antrenament singe-view. În sfârșit, această metodă se bazează în principal, pe termenul de reconstrucție a imaginii, ceea ce înseamnă că suprafețele speculare și transparente vor produce adâncimi inconsistente.

Here we propose a framework for jointly training a single-view depth CNN and a camera pose estimation CNN from unlabeled video sequences. Despite being jointly trained, the depth model and the pose estimation model can be used independently during test-time inference. Training examples to our model consist of short image sequences of scenes captured by a moving camera. While our training procedure is robust to some degree of scene motion, we assume that the scenes we are interested in are mostly rigid, i.e., the scene appearance change across different frames is dominated by the camera motion.

3.8.3 Zhou et al. 2017 antrenarea monoculară

O metodă nesupervizată pentru sarcina de estimare a adâncimii monoculare și a mișcării camerei din secvențe video nestructurate. În comun cu munca ulterioară [10, 14, 16], se folosește o abordare de învățare *end-to-end* cu sinteza de vedere ca semnal de supraveghere. Spre deosebire de lucrările anterioară, metoda dată este complet nesupervizată, necesitând doar secvențe video monoculare pentru antrenare. Metoda dată utilizează rețelele de profunzime *single-view* și rețelele *multi-view*, cu o pierdere bazată pe distorsionarea vederilor din apropiere către țintă utilizând

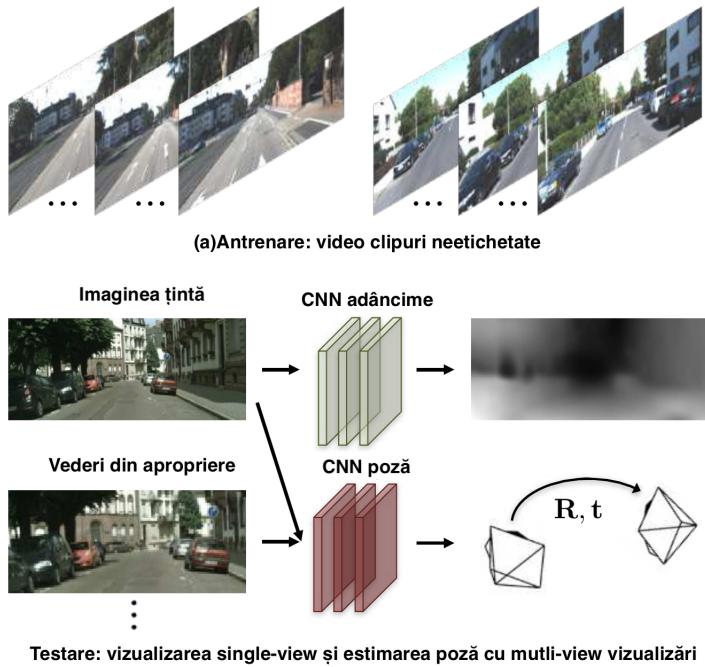


Figure 3.9: The training data to our system consists solely of unlabeled image sequences capturing scene appearance from different viewpoints, where the poses of the images are not provided. Our training procedure produces two models that operate independently, one for single-view depth prediction, and one for multi-view camera pose estimation.

adâncimea și poziția calculată. Astfel, rețelele sunt cuplate cu pierderea, în timpul antrenamentului, dar pot fi aplicate independent în timpul testării.

În această lucrare, problema adâncimii în imagini monoculare este abordată prin formarea unui model, care urmărește secvențe de imagini și are scopul de a explica observațiile sale prin prezicerea mișcării probabile a camerei și a structurii scenei aşa cum este arătat în Figure 3.9. Se realizează o abordare end-to-end, permitând modelului mapeze direct de la pixeli de intrare la o estimare a mișcării ego-ului (parametrizat ca matrice de transformare) și structura de bază (parametrizată ca hărți de adâncime pe pixeli sub imagine de referință).

Abordarea dată se bazează pe cunoașterea faptului că un sistem de sinteză de vizualizare geometrică funcționează numai în mod consistent, atunci când predicțiile sale intermediare ale geometriei scenei și ale poziției camerei corespund punctului fizic. Deși geometria imperfectă și / sau estimarea poză poate să trișeze cu vederi rezonabile sintetizate pentru anumite tipuri de scene (de exemplu fără textură), același model va eșua miserabil atunci când va fi prezentat cu un alt set de scene cu amplasare mai diversă și aparență structurilor. Astfel, obiectivul acestei metode este de a formula întreagul pipeline de sinteză a vederilor ca procedură de deducere a unei rețele neuronale convoluționale, astfel încât prin formarea rețelei pe date video

Metoda	Baza De Date	RMS	logRMS	Absolute relative	Square relative	$\delta < 1.25$
Garg et al.	Kf	5.104	0.273	0.169	1.080	0.740
Godard et al.	CS + K	4.935	0.206	0.114	0.898	0.861
Zhou et al.	CS + K	6.565	0.275	0.198	1.836	0.718

Table 3.3: Comparatia erorilor metodelor nesupervizate

la scară mare pentru meta-sarcina sintezei de vedere rețeaua este forțată să învețe despre sarcinile intermediare de adâncime și despre estimarea pozei aparatului foto pentru a găsi o explicație consistentă a lumii vizuale. Evaluarea empirică pe baza de date KITTI [16] demonstrează eficacitatea abordării date atât în ceea ce privește adâncimea de vizualizare cu o singură perspectivă, cât și estimarea poziției camerei.

3.9 Învățare semi-supervizată

Kuznetsov et al. [11] propune o metoda ce învăță într-un mod semi supervizat. Pentru învățarea supervizată se folosesc hărți de adâncime cu raspindire rară, și se mai impune rețeaua să producă hărți de adâncime photoconsistente într-un amplasare stereofoto, folosind o pierdere directă de aliniere a imaginii.

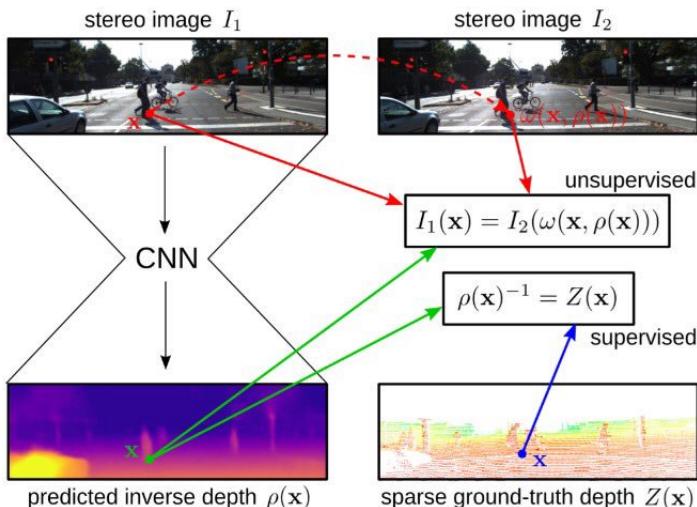


Figure 3.10: Este învățat în mod simultan un CNN de la indicii nesupervizate și supraveghere. Pentru antrenamentul supervizat este folosit citiri de adâncime adevărată rare, cu ajutorul unui dispozitiv, cum ar fi un laser 3D. Alinierea nesupervizată directă a imaginii completează măsurătorile adevărate cu un semnal de antrenament care se bazează exclusiv pe imaginile stereo și pe harta de profunzime precisă pentru o imagine.

Rețeaua noastră este formată din stive cu două componente, prezentate în figura 1. Rețeaua la scară largă prezice profunzimea scenei la nivel global. Acest lucru este apoi rafinat în regiunile locale printr-o rețea la scară redusă. Ambele stive sunt aplicate la intrarea originală, dar în plus, ieșirea rețelei grosiere este trecută la rețeaua fină ca și caracteristici suplimentare de imagine de prim strat. În acest fel, rețeaua locală poate edita predicția globală pentru a include detalii detaliante.

Sa comparat intre metoda propusă de Kuznetsov [11] cu metodele de ultimă generație pe imaginile de testare a KITTI benchmark. Metoda dată depășește cel mai bun setup al lui Gordar et al. [41] cu 1.16 m circa 14 %, in termenii RMSE, si de 0.035 circa 16 % pentru scala ei log capăt la 80 m.

3.10 Stereo vision de ultimă generație Deep3D

Xie et al. 2016 nu încearcă să rezolve problema adâncimii, propune un algoritm automat de conversie 2D-la-3D care ia imagini 2D sau imagini video ca input și dă la output perechi de imagini stereo 3D. Adică, e propus să regreseze direct imaginea dreaptă din imaginea stîngă, cu pierdere pe pixel-wise. Dar abordarea naivă a acestei metode, duce spre rezultate slabe. Pentru a îmbunătăți aceste rezultate, se utilizează un DIRP (Depth image based rendering) pentru a captura faptul că majoritatea pixelii de output sănt copii de săftări a pixelor de input. Sistemul nu este constrâns să producă hărți de adâncime, nici nu trebuie hărți de adâncime ca antrenare.

Propune un model care prezice o hartă asemănătoare probabilității disparităților ca un output intermedian și o combinație cu vizualizarea de input folosind un strat de selecție diferențiat care modelează procesul DIBR. În timpul antrenamentului, hărțile asemănătoare a disparității produse de model nu sunt niciodată comparate direct cu o hartă reală a disparității și se termină cu servirea scopurilor duale de reprezentare a disparității orizontale și a realizării in-painting.

3.11 2017 perspectivă sintetizată(*view synthesis*) + potrivirea stereo (*stereo matching*)

Luo et al. [18] reformează problema adâncimii ca două sub-probleme, procedură a perspectivei sintetizată(view synthesis), și potrivirea stereo (stereo matching) cu două proprietăți: 1) constrângerile geometrice pot fi impuse la inferență, 2) cererea privind datele de adâncime etichetate poate fi mult ameliorată. Această nouă formulare, cu un pipeline end-to-end joacă un rol important în avansarea performanței. Prin abordarea problemei în acest mod, ambele proceduri se supun principiilor principale geometrice, și pot fi antrenate fără furnizarea de date scumpe. Mulțumint antrenamentului comun(joint), performanța pipelinului este promovată.

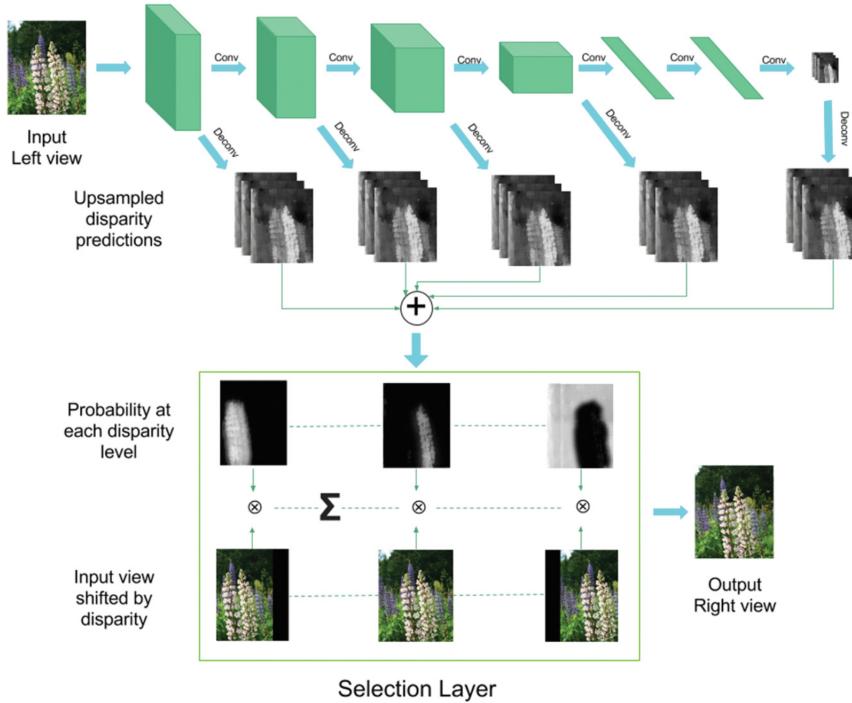


Figure 3.11: Architectura metodei

Pentru perspectiva sintetizată este propusă, din Deep3D [13], o nouă schemă probabilistică pentru a transera pixelii din imaginea originală, ea formulează direct transformarea din imaginea stângă în imaginea dreaptă folosind un strat de selecție diferențiat. După aceasta, rețeaua de potrivirea stereo transformă problema de nivel înalt a inteligenței scenei într-o problemă de potrivire stereo 1D. Pentru a utiliza mai bine relația geometrică dintre aceste două imagini, este folosit DispNetC[14], și pentru a obține predicția totală a rezoluției ei adoptează DispFullNet menționată în [15].

Rezultate În comparație cu metodele de ultimă generație, metoda dată depășește pe aproape toate metricile. Cu metricile ARD reduse cu 17.5% în comparație cu Gorard et al. [41] și 16.8 % în comparație cu Kuznetsov et al.[11] cu capat de 80 m. Antrenarea end-to-end optimizează colaborare celor două sub rețele ce aduce la rezultate de ultima generație.

Fiind un algoritm de ultimă generație, fără a folosi o cantitate mare de etichete scumpe cu adâncimea adevărată, modelul dat performă mai bine decât toate metodele precedente de estimare a adâncimei monoculară. și sănă primii care au depășit algoritmul *stereo blocking matching* pe un benchmark de potrivire stereo folosind o metodă monoculară.

3.12 Eroarea și exemple

Cele mai bune rezultate din punct de vedere a errorii apar în ultima lucrare de Luo et al. ce a implementat o rețea compusă din două subprobleme, sintetizarea unei imagini din alta, și potrivirea stereo. După

Diferența relativă absolută (ARD): $\frac{1}{|T|} \sum_{y \in T} |y - y^*| / y^*$

Diferența pătratică relativă (SRD): $\frac{1}{|T|} \sum_{y \in T} |y - y^*|^2 / y^*$

relative (rel) error = $\frac{|D - D^*|}{D^*}$, D - adâncimea estimată D^* - adâncimea adevărată

$\log_{10} = |\log(D) - \log_{10}(D^*)|$,

RMSE = $\sqrt{\frac{1}{N} \sum_{i=1}^N \|Dep_i - Dep_i^{g.t.}\|^2}$,

RMSE(log) = $\sqrt{\frac{1}{N} \sum_{i=1}^N \|\log(Dep_i) - \log(Dep_i)^{g.t.}\|^2}$,

Metoda	Baza De Date	RMSE	RMSE(log)	ARD	SRD	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Saxena et al. [1]	Make3D	-	0.187	0.370		0.447	0.745	0.897
	NYU	1.214	0.409	0.349	0.492	0.447	0.745	0.897
	KITTI	8.734	0.361	0.280	3.012	0.601	0.820	0.926
Karsch et al. [6]	Make3D	15.1	0.148	0.361				
	NYU	1.2		0.350				
Ladicky et al. [5]	NYU					0.542	0.829	0.941
Eigen et al. [9]	NYU	0.907	0.285	0.215	0.212	0.611	0.887	0.971
	KITTI	7.156	0.270	0.190	1.515	0.692	0.899	0.967
Liu et al. [34]	NYU	1.08	0.126	0.327				
	Make3D	12.6	0.134	0.338				
Li et al. [7]	NYU	0.759	0.091	0.223		0.639	0.900	0.974
	Make3D	10.27	0.102	0.279				
Laina et al. [8]	NYU	0.573	0.195	0.127		0.811	0.953	0.988
	Make3D	4.46		0.176				
Chen et al. [37]	NYU	1.13	0.39	0.360	0.460			
Kuznetsov et al. [11]	KITTI 1-50 M	3.518	0.179	0.108	0.595	0.875	0.964	0.988
	KITTI 1-80 M	4.621	0.189	0.113	0.741	0.862	0.960	0.986
Luo et al. [18]	KITTY 1-50 M	3.266	0.167	0.090	0.499	0.902	0.968	0.986
	KITTY 1-80 M	4.252	0.177	0.094	0.626	0.891	0.968	0.986

Table 3.4: (NYU)NYU Depth v2 [23], Make3D [3, 2], KITTI [16], ARD - diferență absolută relativă, diferență pătratică relativă - SRD

4 Percepția adâncimii dintr-o singură imagine exemplu

Xu et al. 2015 mai întâi generează sistemul de coordonate de adâncime a planului de pămînt dintr-o singură imagine monoculară prin principiul de formare a imaginii și apoi localizează obiectele în imagine cu sistemul de coordonate utilizând caracteristicile geometrice. Adică, se oferă o metodă de estimare a hărților de adâncime exacte. Această lucrare se concentrează asupra imaginilor care conțin solul, deoarece planul de la sol va fi un cadru de referință superior al adâncimii și majoritatea scenelor din imagini includ planul de bază.

În ultimul zeci de ani, au fost propuse mai multe metode de estimare a adâncimilor de la imagini monoculare, dar aproape toate metodele care se bază pe Markov Random Field (MRF) sunt sensibile la obiectele multicolore dintr-o imagine și au nevoie de mulți parametri de învățare, adică sănătatea este dificil de învățată. Unele metode care utilizează caracteristici geometrice estimatează adâncimea relativă a obiectelor sau a scenei dintr-o imagine, dar nu își pot obține profunzimea absolută.

Pentru a rezolva problemele de mai sus, se propune o abordare bazată pe o poziție de formare a imaginii și să ia în considerare caracteristicile geometrice. Metoda dată are nevoie de mai puțini parametri, în timp ce poate obține informațiile absolute de adâncime și poate reduce efectele obiectelor multicolore în imagini. Se bazează pe presupunerea că obiectele nu sunt artărnate în aer și adâncimea unui obiect poate fi determinată de punctele pe care le atinge pe planul de la sol sau alte obiecte. De asemenea, presupunem că axa optică a dispozitivului de formare a imaginii este întotdeauna paralelă cu planul de la sol.

4.1 Ground Plane Depth Coordinate System

După cum știm, relația dintre lumea reală 3D și imaginile digitale 2D poate fi descrisă printr-un sistem de coordonate. În abordarea dată, se analizează mai întâi principiul de formare a imaginii și apoi se propune o metodă de generare a sistemului de coordonate de adâncime a planului de la sol, pe imagini.

4.1.1 Principiul de Formalre a Imaginei

În fig. 1, punctul O este poziția lentilei dispozitivului de formare a imaginea. Acum se folosește sistemul de coordonate de profunzime $X_C Y_C Z_C$ pentru a descrie spațiul real al lumii. În acest sistem de coordonate, un punct $P(X, Y, Z)$ în lumea reală este transformat într-un punct 2D $p(x, y)$ pe planul de imagine real. În principiul de formare a imaginii, dispozitivul primește lumină din exterior și formează o imagine, cu capul în jos pe planul de imagine real din spatele lentilei cu distanța F , care se numește distanță focală. Pentru a ajuta la înțelegere, se adaugă un plan de imagine virtual în fața obiectivului cu o distanță F . O imagine cu orientarea corectă va fi formată pe planul de imagine virtual. Evident, este imposibil de a calcula valoarea exactă a adâncimii dintre cele două sisteme de coordonate diferite, dar putem obține relația lor care este utilă pentru estimarea adâncimii.

În condiția din Fig. 1, noi putem dovedi $\Delta OO'p$ este similar cu $\Delta OO''P$ și $\Delta O'pq$ este similar cu $\Delta O''PQ$. Atunci noi obținem relația în Formula (1).

$$\frac{OO'}{OO''} = \frac{O'p}{O''p}, \frac{O'p}{PQ} = \frac{pq}{PQ} = \frac{O'p}{O''Q}$$

$$\frac{F}{Z} = \frac{y}{Y} \quad (4.1)$$

Formula (Equation 4.1) arată relația dintre coordonatele punctelor 2D și cele ale punctelor 3D, care ajută să ne dăm seama cum schimbarea adâncimii în lumea reală reacționează asupra imaginii.

Se presupune că există două puncte $P_0(X, Y, Z_0)$ și $P_1(X, Y, Z_1)$ în sistemul 3D de coordonate din lumea reală și distanța dintre cele două puncte pe axa Z_C este d . În Fig 2, punctul P_0 este mapat la $p_0(x_0, y_0)$ pe planul de imagine și punctul P_1 este mapat la $p_1(x_1, y_1)$. Pe baza formulei (Equation 4.1), se poate obținut formula Equation 4.2 și relația dintre coordonatele 2D ale punctelor și adâncimea lor în lumea reală ca Formula (Equation 4.3).

$$\frac{F}{Z_0} = \frac{y_0}{Y}, \frac{F}{Z_1} = \frac{y_1}{Y} \quad (4.2)$$

$$\frac{y_1}{y_0} = \frac{Z_0}{Z_1} \quad (4.3)$$

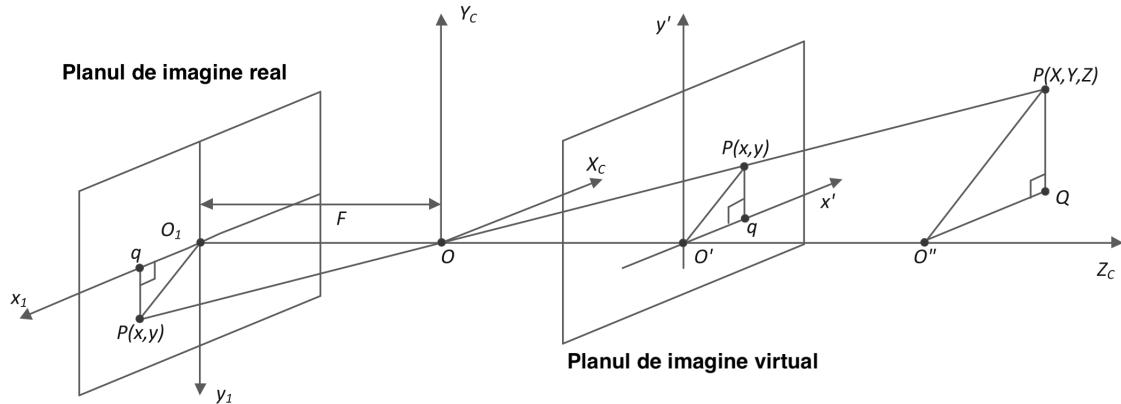


Figure 4.1: Principiul de formare al imaginii optice

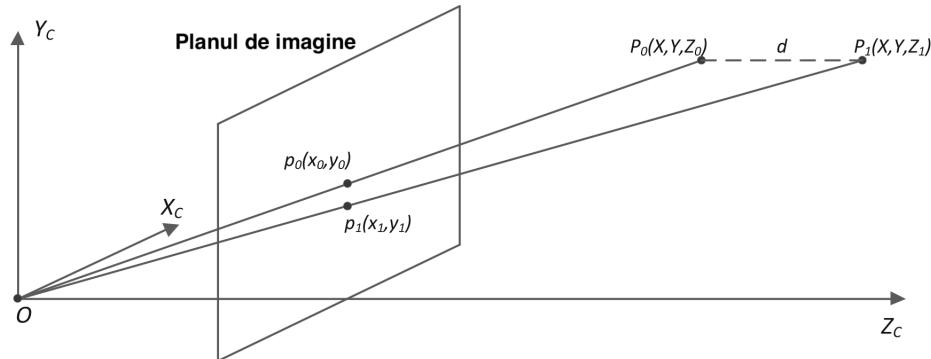


Figure 4.2: Relația dintre coordonatele 2D și 3D

4.2 Algoritmul de Generare a Sistemului de Coordonate de Adâncime

Aici este propusă o metodă de generare a sistemului de coordonate de adâncime. În Formula (Equation 4.3), este oferită relația dintre coordonatele celor două puncte 2D și profunzimea lor în lumea reală. Din această relație, se poate deduce că toate punctele de pe planul de la sol din imagine au o relație similară. Deci, se poate folosi relația pentru a genera sistemul de coordonate de adâncime al planului de sol în imagine.

Se presupune că există un set P , care conține n puncte în lumea reală 3D. Aceste puncte au întotdeauna aceleași coordonate pe axa X_C și Y_C și la fiecare două puncte învecinate $P_i(X, Y, Z_i)$ și $P_{i+1}(X, Y, Z_{i+1})$ au aceeași distanță d (de fapt,

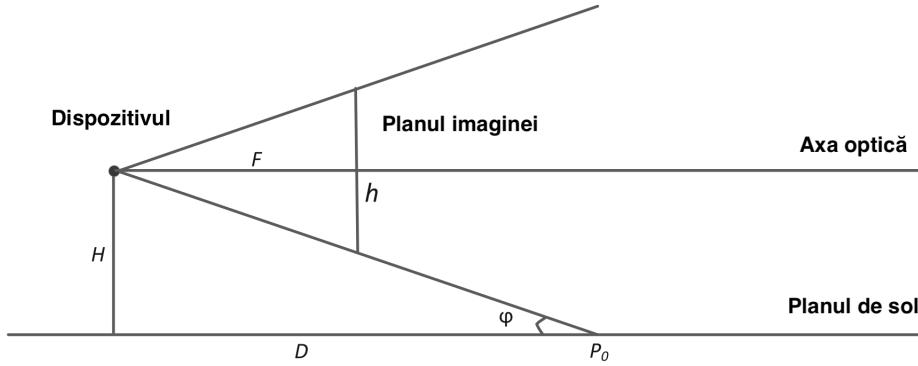


Figure 4.3: Caracteristicile dispozitivului

aceste puncte aparțin aceleiași linii, iar linia este pe planul de sol și paralel cu axa optică a dispozitivului). Se poate reprezenta sistemul de coordonate de adâncime pe imagine folosind un set C.

$$C = \{y_i \mid P_i \in P, i = 0, 1, \dots, n\} \quad (4.4)$$

Din concluzia precedentă din Formula (Equation 4.3) și Formula (Equation 4.4), nu e dificil să se propună metoda de generare a sistemului de coordonate de adâncime. Dar altă problemă apare aici, cum să se determine adâncimea primului punct P_0 ($P_0 \in P$).

În Fig3, distanța verticală de la dispozitiv pînă la pămînt este H . Înălțimea planului imaginii este h , iar distanța dintre dispozitiv și planul de imagine este distanța focală F . Unghiul dintre intervalul vizual și axa optică este reprezentat ca φ . Se presupune că adâncimea celui mai de jos rând al imaginii de pe planul de la dispozitiv de imagine este D . Deci, putem obține expresia D ca Formula (Equation 4.5).

$$D = \frac{H}{\tan \varphi} = \frac{2FH}{h} \quad (4.5)$$

Din formula (Equation 4.5), D este afectată numai de distanța focală F , de altitudinea H a dispozitivului și altitudinea h a planului de imagine. Atunci când se procesează imaginile vizuale robotice sau imaginile cu parametrii dispozitivului, putem obține cu ușurință aceste stări pentru a calcula valoarea lui D . De asemenea, putem utiliza câteva valori experimentate pentru a determina valoarea lui D dacă nu există informații despre starea dispozitivului. Valoarea lui $D \in [5, 10]$ (metri) poate fi folosită în majoritatea imaginilor de pe Internet sau în date seturi. Apoi

oferez o formulă de recurență în cazul general, după cum urmează.

$$y_{n+1} = \left(\frac{D + nd}{D + (n + 1)d} \right) y_n \quad (4.6)$$

Algoritm 1.

Table 4.1:

Algoritmul 1. Algoritmul de generare a sistemului de coordonate de adâncime

Input:	Caracteristicile parametrilor ai dispozitivului h, F; Parametrii de stare ai dispozitivului H; Adâncimea unității d și setul de puncte P.
Output:	Setul de coordonate C.
Pas 1:	Calculați distanța orizontală D prin formula (5).
Pas 2:	Setați $C = \emptyset$, $i = 0$, $y_0 = -\frac{ROW}{2}$, pune y_0 în setul C. (ROW este rândurile totale ale imaginii prelucrate.)
Pas 3:	Utilizați y_i pentru a calcula coordonatele punctului următor $P_{i+1} \in P$ cu o adâncime a unității, y_{i+1} , prin formula (Equation 4.6), apoi puneti y_{i+1} în setul C
Pas 4:	Dacă avem următorul punct $P_{i+2} \in P$, setăm $i = i + 1$ și mergeți la pasul 3; altfel, mergeți la pasul 5.
Pas 5:	Returnați setul de coordonate C.

Prin Algoritmul 1, obținem setul de coordonate C , care conține toate coordonatele 2D, date de puncte 3D pe planul solului cu o adâncime diferită. Putem obține diferite coordonate la scară precisă prin reglarea adâncimii unității d . În Fig4, am setat adâncimea unității $d = 2(m)$ a coordonatei și am setat-o pe cea mijlocie ca $d = 1(m)$, cea dreaptă ca $d = 0.5(m)$. Performanța arată că cu cât este mai mică d , cu atât coordonatele sunt mai exacte. Dar stabilirea valorii d prea mică nu este necesară și duce spre ineficientă. Pe baza unui număr mare de experimente, se constată că $d = 1(m)$ poate obține o performanță bună și păstrează o eficiență ridicată.

4.3 Algoritm de percepție a adâncimii în imagine

Pentru a determina adâncimea fiecărui punct dintr-o imagine, ar trebui să înțelegem mai întâi diferențele zone dintr-o imagine. Folosim un algoritm de segmentare a texturii bazat pe [19, 20, 21], care utilizează caracteristicile geometrice. Apoi propunem un algoritm de percepție a adâncimii pentru a obține harta de adâncime a imaginii pe baza rezultatului de preprocesare a segmentării.

4.3.1 Segmentarea de Textură

Deoarece sistemul de coordonate de adâncime este utilizat pentru maparea planului de la sol la o adâncime diferită, în primul rând imaginile trebuie să fie preprocesate pentru a obține zona de pământ. Pe baza caracteristicilor geometrice, obiectele din imagini sunt categorizate exact [19], unde cerul, solul și verticala în imagini sunt împărțite în zone. În fig5, imaginea A este imaginea originală, imaginea B este rezultatul clasificării. În imaginea B, partea albastră este zona de cer, cea verde este suprafața solului, iar cea roșie este zona verticală. Săgețile "↑", "←" sau "→" pe obiecte arată direcția de schimbare a adâncimii. Marcile "?" Sau "x" pe obiecte reprezintă faptul că aceste obiecte sunt solide sau goale, respectiv.

După ce cerul, solul și zonele verticale au fost separate, împărțim cu mai multă precizie [20, 21] pentru a obține locația obiectelor (a se vedea Fig5, imaginile B și C).

4.3.2 Percepția adâncimii în imagine

În această subsecțiune, se propune o metodă de percepție a adâncimii în imagine, care include percepția de adâncime a planului de la sol și percepția adâncimii a zonei verticale. Din acest motiv, putem stabili adâncimea cerului infinit.

După preprocesare, se obține o imagine în care fiecare obiecte și zone sunt separate una de celalătă. Orice părți ale rezultatului de segmentare pot fi descrise ca un quad de R_i (C_i, M_i, B_i, P_i). Acest quad furnizează atributele și informațiile de localizare ale unei regiuni de segmentare. În acest quad, ambele C_i și M_i reprezintă categorii ale regiunii. C_i arată regiunea care aparține cerului, planului de la sol sau verticală, iar M_i arată ce fel de marcă este în regiune. B_i este numărul liniei de bază a regiunii. P_i este coordonata 2D a punctului din stânga sus al regiunii.

Se presupune că un set R include toate regiunile imaginii segmentate. setul R poate fi descris ca $R = \{R_i \mid i = 0, 1, \dots, n\}$, unde n este numărul total de regiuni dintr-o imagine.

Prin presupunerea că nu sunt atârnate obiecte în aer și adâncimea unui obiect poate fi determinată de punctele pe care le atinge planul de la sol, putem estima cu ușurință informațiile despre profunzimea a majorităților de regiunilor. Dar există anumite regiuni acoperite de alte regiuni sau care nu ating direct planul de la sol, deci este dificil să se estimateze informațiile despre adâncime. Deoarece lucrările din [12] arată că regiunile învecinate pot oferi indicii importante pentru a deduce informațiile despre adâncime dintr-o regiune. Inspirat de [12], se propune o metodă de estimare a informațiilor privind adâncimea regiunii de către regiunile învecinate, pe baza lungimii lor de atingere. Fiecare vecin care și-a stabilit adâncimea ar trebui să contribuie la stabilirea profunzimii acestei regiuni. Utilizăm următoarea ecuație

pentru a determina adâncimea regiunilor care nu ating direct pământul:

$$dep = \sum_{j=1}^m r_j dep_j \quad (4.7)$$

În formula (Equation 4.7), m este numărul total de regiuni învecinate care au fost determinate adâncimea lor, r_j este lungimea de atingere celor două regiuni. Valoarea din r_j este dat de $r_j = \frac{TL_j}{\sum_{j=1}^m TL_j}$. TL_j în ecuație înseamnă lungimea atingerii a două regiuni. Între timp, trebuie să satisfacă $\sum_{j=1}^m r_j = 1$.

Table 4.2:

Algoritmul 2. Percepția adâncimei a imaginii.	
Input:	Setul de coordonate C ; Setul de regiune $R. (R = \{R_i(C_i, M_i, B_i, P_i) i = 1, 2, \dots, n\})$
Output:	Harta de adâncime a imaginii.
Pas 1:	Setează un set ca $RF = \emptyset$, Inițializează harta de adâncime goală.
Pas 2:	Traversează R , găsește toate $R_i \in R$, unde $C_i = SKY$. Setează adâncimea pentru fiecare R_i valoare maximă, sterge R_i din R și punele în RF .
Pas 3:	Traversează R , găsiți toți $R_i \in R$, unde $C_i = GROUND$. Setați adâncimea fiecărei R_i pentru valoarea corespunzătoare pe baza setului de coordonate C , scoateți R_i din R și puneti-l în RF .
Pas 4:	Scanați celelalte elemente în R și găsiți $R_i \in R$ care atinge planul de sol și are minimum B_i . Dacă reușește, mergeți la pasul 5; altfel, mergeți la pasul 6.
Pas 5:	În conformitate cu M_i și punctele de atingere ale R_i , setați adâncimea R_i , eliminați R_i de la R și puneti-l în RF . Mergeți la pasul 4.
Pas 6:	Scanați celelalte elemente în R și găsiți $R_i \in R$ care are minimum B_i . Dacă reușește, mergeți la pasul 7; altfel, mergeți la pasul 8.
Pas 7:	Calculați adâncimea bazei R_i cu Formula (Equation 4.7), setați adâncimea R_i , stergeți R_i de R și puneti-l în RF . Mergeți la pasul 6.
Pas 8:	Returnează harta de adâncime.

5 Concluzie

Scopul tezei date a fost de a face un studiu de caz, pe problema percepție adâncimii dintr-o imagine monoculară, și de a găsi algoritmele de ultimă generație, cu cerințe de hardware minime, timpul antrenamentului, viteza mică de testare pentru procesarea secvențelor de imagini, mărimea datelor de antrenament. Înainte de a concepe un model, apare problema datelor de antrenament. Date de acest fel sunt foarte greu de crea, deoarece extragerea adâncimii dint-un peisaj e o problemă de hardware complicată, o metodă este folosirea unui dispozitiv laser, dar din cauza suprafetei strălucitoare, nealinierea dispozitivului laser cu cel al camerei, peisaj în mișcare, și rezoluția mică a hărții de adâncime, duce spre a crea date cu posibilitatea de a avea erori. O altă metodă este de a folosi date de antrenament în formă de *stereo vision*. Dar hărți de disparitate produc hărți de adâncime relative, cu limitări la distanțe mari și foarte mici și informația de adâncime absolută este mai ambiguă. Unele metode au încercat să creeze rețele nesupervizate învățate cu date neetichetate.

Bibliography

- [1] Saxena, Ashutosh, Min Sun, and Andrew Y. Ng. "Make3D: Depth Perception from a Single Still Image." AAAI. 2008.
- [2] Saxena, Ashutosh, Min Sun, and Andrew Y. Ng. "Make3d: Learning 3d scene structure from a single still image." IEEE transactions on pattern analysis and machine intelligence 31.5 (2009): 824-840.
- [3] Saxena, Ashutosh, Sung H. Chung, and Andrew Y. Ng. "Learning depth from single monocular images." Advances in neural information processing systems. 2006.
- [4] Eigen, David, and Rob Fergus. "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture." Proceedings of the IEEE International Conference on Computer Vision. 2015.
- [5] Ladicky, Lubor, Jianbo Shi, and Marc Pollefeys. "Pulling things out of perspective." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014.
- [6] Karsch, Kevin, Ce Liu, and Sing Bing Kang. "Depth extraction from video using non-parametric sampling." European Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012.
- [7] Liu, M., Salzmann, M., & He, X. (2014). Discrete-Continuous Depth Estimation from a Single Image. 2014 IEEE Conference on Computer Vision and Pattern Recognition, 716–723. <https://doi.org/10.1109/CVPR.2014.97>
- [8] Laina, Iro, et al. "Deeper depth prediction with fully convolutional residual networks." 3D Vision (3DV), 2016 Fourth International Conference on. IEEE, 2016.
- [9] Eigen, David, Christian Puhrsch, and Rob Fergus. "Depth map prediction from a single image using a multi-scale deep network." Advances in neural information processing systems. 2014.
- [10] Godard, Clément, Oisin Mac Aodha, and Gabriel J. Brostow. "Unsupervised monocular depth estimation with left-right consistency." CVPR. Vol. 2. No. 6. 2017.
- [11] Kuznetsov, Yevhen, Jörg Stückler, and Bastian Leibe. "Semi-supervised deep learning for monocular depth map prediction." Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. 2017.

- [12] Ladicky, Lubor, Jianbo Shi, and Marc Pollefeys. "Pulling things out of perspective." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014.
- [13] Xie, Junyuan, Ross Girshick, and Ali Farhadi. "Deep3d: Fully automatic 2d-to-3d video conversion with deep convolutional neural networks." European Conference on Computer Vision. Springer, Cham, 2016.
- [14] Mayer, Nikolaus, et al. "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. 2016.
- [15] J. Pang, W. Sun, J. S. Ren, C. Yang, and Q. Yan. Cascade residual learning: A two-stage convolutional neural network for stereo matching. arXiv preprint arXiv:1708.09204, 2017
- [16] Geiger, Andreas, Philip Lenz, and Raquel Urtasun. "Are we ready for autonomous driving? the kitti vision benchmark suite." Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, (KITTI), 2012.
- [17] Cordts, Marius, et al. "The cityscapes dataset for semantic urban scene understanding." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
- [18] Luo, Yue, et al. "Single view stereo matching." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.
- [19] Hoiem, Derek, Alexei A. Efros, and Martial Hebert. "Geometric context from a single image." Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on. Vol. 1. IEEE, 2005.
- [20] Felzenszwalb, Pedro F., and Daniel P. Huttenlocher. "Efficient graph-based image segmentation." International journal of computer vision 59.2 (2004): 167-181.
- [21] Hoiem, Derek, Alexei A. Efros, and Martial Hebert. "Closing the loop in scene interpretation." Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008.
- [22] Firman, Michael. "RGBD datasets: Past, present and future." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2016.
- [23] Silberman, Nathan, et al. "Indoor segmentation and support inference from rgbd images." European Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012.
- [24] Menze, Moritz, and Andreas Geiger. "Object scene flow for autonomous vehicles." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

Bibliography

- [25] Menze, M., Ch Heipke, and A. Geiger. "Joint 3d estimation of vehicles and scene flow." ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences 2 (2015): 427.
- [26] Mayer, Nikolaus, et al. "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. 2016.
- [27] Liu, Ce, Jenny Yuen, and Antonio Torralba. "Nonparametric scene parsing: Label transfer via dense scene alignment." Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009.
- [28] Vedaldi, Andrea, and Karel Lenc. "Matconvnet: Convolutional neural networks for matlab." Proceedings of the 23rd ACM international conference on Multimedia. ACM, 2015.
- [29] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
- [30] Zwald, Laurent. "The BerHu penalty and the grouped effect Sophie Lambert-Lacroix UJF-Grenoble 1/CNRS/UPMF/TIMC-IMAG UMR 5525, Grenoble, F-38041, France and." arXiv preprint arXiv:1207.6868 (2012).
- [31] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385, 2015.
- [32] Dosovitskiy, Alexey, Jost Tobias Springenberg, and Thomas Brox. "Learning to generate chairs with convolutional neural networks." Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on. IEEE, 2015.
- [33] F. Liu, C. Shen, and G. Lin. Deep convolutional neural fields for depth estimation from a single image. In Proc. Conf. Computer Vision and Pattern Recognition (CVPR), pages 5162–5170, 2015.
- [34] Li, Bo, et al. "Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.
- [35] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012.
- [36] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).
- [37] Chen, Weifeng, et al. "Single-image depth perception in the wild." Advances in Neural Information Processing Systems. 2016.
- [38] El-laithy, Riyad A., Jidong Huang, and Michael Yeh. "Study on the use of Microsoft Kinect for robotics applications." Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION. IEEE, 2012.

- [39] Hassner, Tal, and Ronen Basri. "Example based 3D reconstruction from single 2D images." Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on. IEEE, 2006.
- [40] Garg, Ravi, et al. "Unsupervised cnn for single view depth estimation: Geometry to the rescue." European Conference on Computer Vision. Springer, Cham, 2016.
- [41] Godard, Clément, Oisin Mac Aodha, and Gabriel J. Brostow. "Unsupervised monocular depth estimation with left-right consistency." CVPR. Vol. 2. No. 6. 2017.