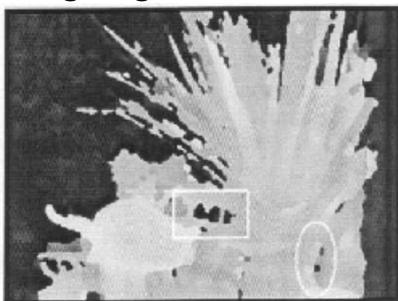


Genetic Based Stereo Algorithm and Disparity Map Evaluation

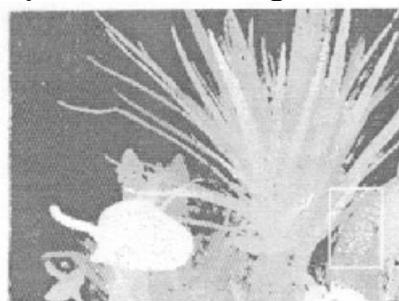
(Vortrag von Alexander Bucksch)

Ziel: Verbesserung der Disparity Map Genauigkeit, unter Verwendung der Ergebnisse anderer Stereo Algorithmen.

Eingangsdaten



optimiertes Ergebnis



Inhalt:

- 1.) Allgemeiner Überblick über den Algorithmus
- 2.) Initialisierung des Disparitätenraumes
- 3.) Codierschema für Disparitätenkarten
- 4.) Fitness Evaluation (Eignungsauswertung)
- 5.) Erzeugung der Initialpopulation
- 6.) Crossover Operator
- 7.) Mutations Operator
- 8.) Minimierungsprozess
- 9.) Parameter Free Measures
- 10.) Beispiele

1. Allgemeiner Überblick über die Vorgehensweise des Algorithmus

- Der Disparitätenraum wird mit dem Ergebnissen der occlusion detection function des SEA (Stereo by eye array) Algorithmus populiert.
- Bilden einer Quadtree Struktur zur Repräsentation der Disparitätenkarte.
- Zur Evaluation des Disparitätenraumes wird eine fitness function (Eignungsfunktion) verwendet.
- Unter Verwendung der fitness function (Eignungsfunktion) als Kriterium, wird ein genetischer Algorithmus benutzt um eine möglichst korrekte Disparitätenkarte zu extrahieren.

2. Initialisierung des Disparitätenraumes

Für zwei Bilder:

$$\begin{aligned} S(r,c,d) &= e_{ref}(r,c,d) \\ &= \sum_{-w \leq i, j \leq w} p(I(r+i, c+j), I_{ref}(r+i, c+j+d)) \end{aligned}$$

w: Fenster radius
p: color dissimilarity function
I(x,y) : Farbe des Pixels an der Stelle (x,y)
e_{ref} : Verschiedenartigkeit/Unterschiedlichkeit
zwischen korrespondierenden Pixeln

für p sollte eine Funktion verwendet werden, die kleine Werte für korrekte Pixelzuordnungen liefert.

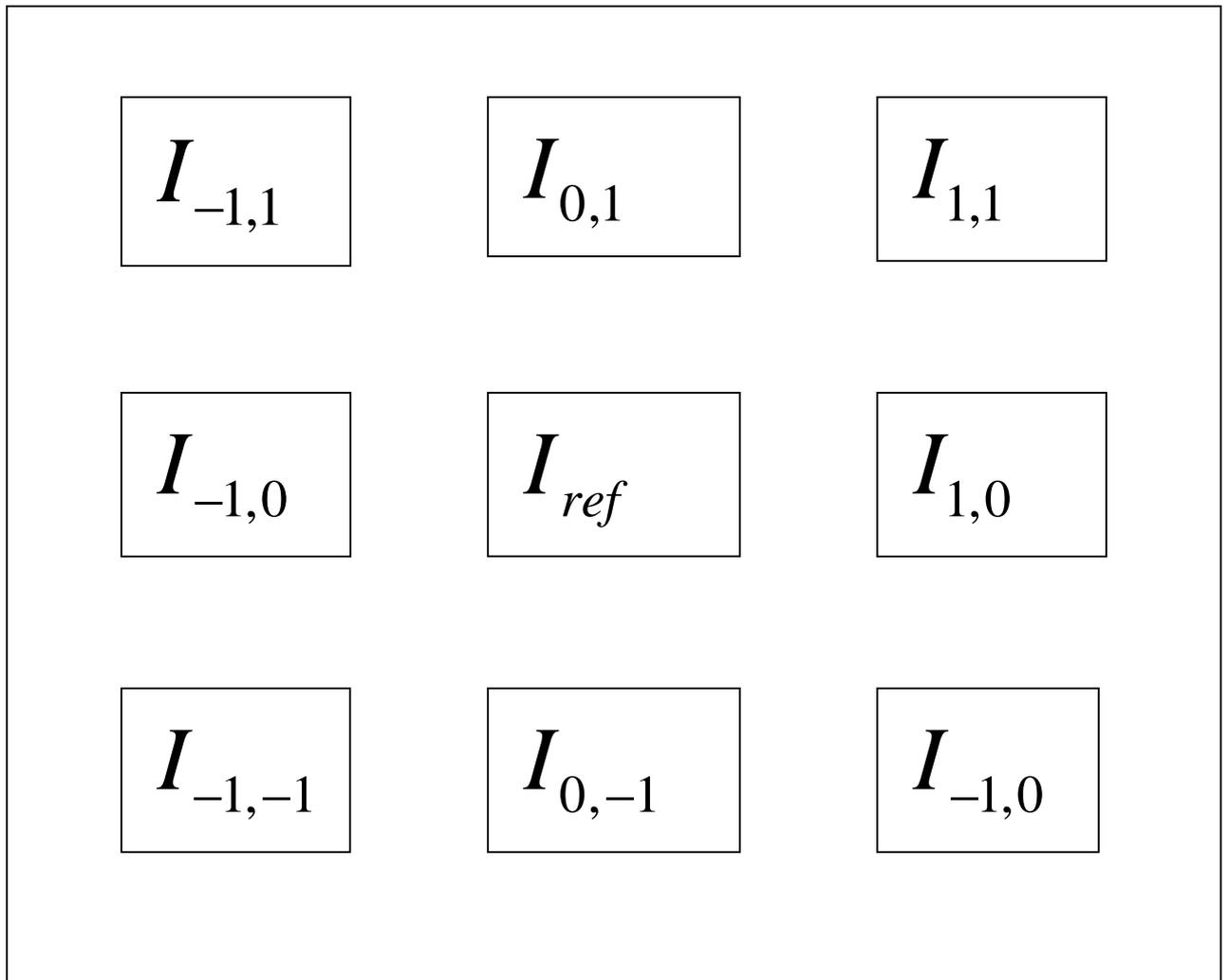
Für multi-view Stereobilder:

$$S(r,c,d) = \sigma \begin{pmatrix} e_{(-1,-1)} & e_{(-1,0)} \\ e_{(-1,1)} & e_{(0,-1)} \\ e_{(0,1)} & e_{(1,-1)} \\ e_{(1,0)} & e_{(1,1)} \end{pmatrix}$$

σ : occlusion detection function

Für σ gibt es einige verschiedene Funktionen. z.B. wurde ein systematischer Ansatz von Satho und Otha gegeben

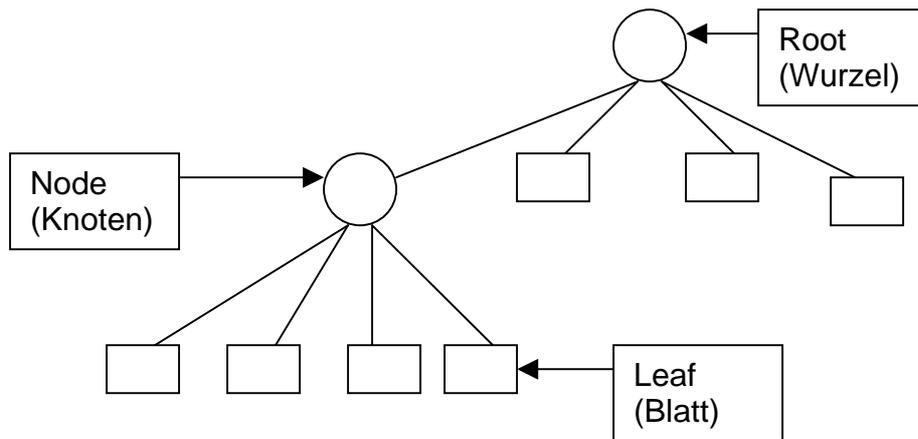
Kameraanordnung des SEA Algorithmus:



I ist ein Bild der entsprechenden Kamera $I_{x,y}$

3. Codier Schema für Disparitätenkarten

- Nutzung der Quadtree Struktur



Dabei gelten zwei zwingende Bedingungen:

1. Jedes Blatt k des Quadtree besitzt einen Disparitätswert
2. Knoten können nicht vier Blätter mit gleichem Disparitätswert haben. In diesem Fall werden alle Blätter gelöscht und aus dem Knoten wird ein Blatt

Ein Quadtree kann nun durch einen eindimensionalen Integerwert repräsentiert werden unter Anwendung der Codiervorschrift:

$$k = \left(\sum_{i=0}^{h-1} 4^i \right) + y \times 2^h + x$$

h : Höhe

x, y : Koordinaten

$$D[k] = \begin{cases} x & \text{,wenn } k \text{ ein Blatt ist und seine Disparität } x \text{ ist} \\ -1 & \text{,sonst} \end{cases}$$

4. Fitness Evaluation (Eignungsauswertung)

Zur Auswertung wird eine Energiefunktion verwendet, welche auf Markov-Zufallsfeldern basiert.

Einschub:

Markov-Zufallsfelder sind Felder in denen jedes Element ein zufälliger Wert ist. Zwischen den einzelnen Elementen bestehen jedoch Beziehungen.

$$f(D) = \sum_{k \in P} \left(\sum_{(r,c) \in k} S(r,c, D[k]) + \lambda T_k \right)$$

$$T_k = \sum_{g \in P \cap Q_k} (1 - \delta(D[k] == D[g])) \times t_{g,k}$$

P: alle Blätter des Quadrees

D[k]: Disparitätswert eines Blattes k

λ : Wichtung Strafterms

T_k: Länge der Kante von Blatt k

δ : boolescher Wert 1 oder 0

Q_k: alle Knoten, die eine gemeinsame Grenze mit k haben.

t_{g,k}: Länge der gemeinsamen Grenze von Blatt k und g

5. Erzeugung der Initialpopulation

- Genetische Algorithmen benötigen eine Initialpopulation als Startwert.
- Dieser Prozess ist im hier vorgestellten Algorithmus rein zufällig

Ablauf:

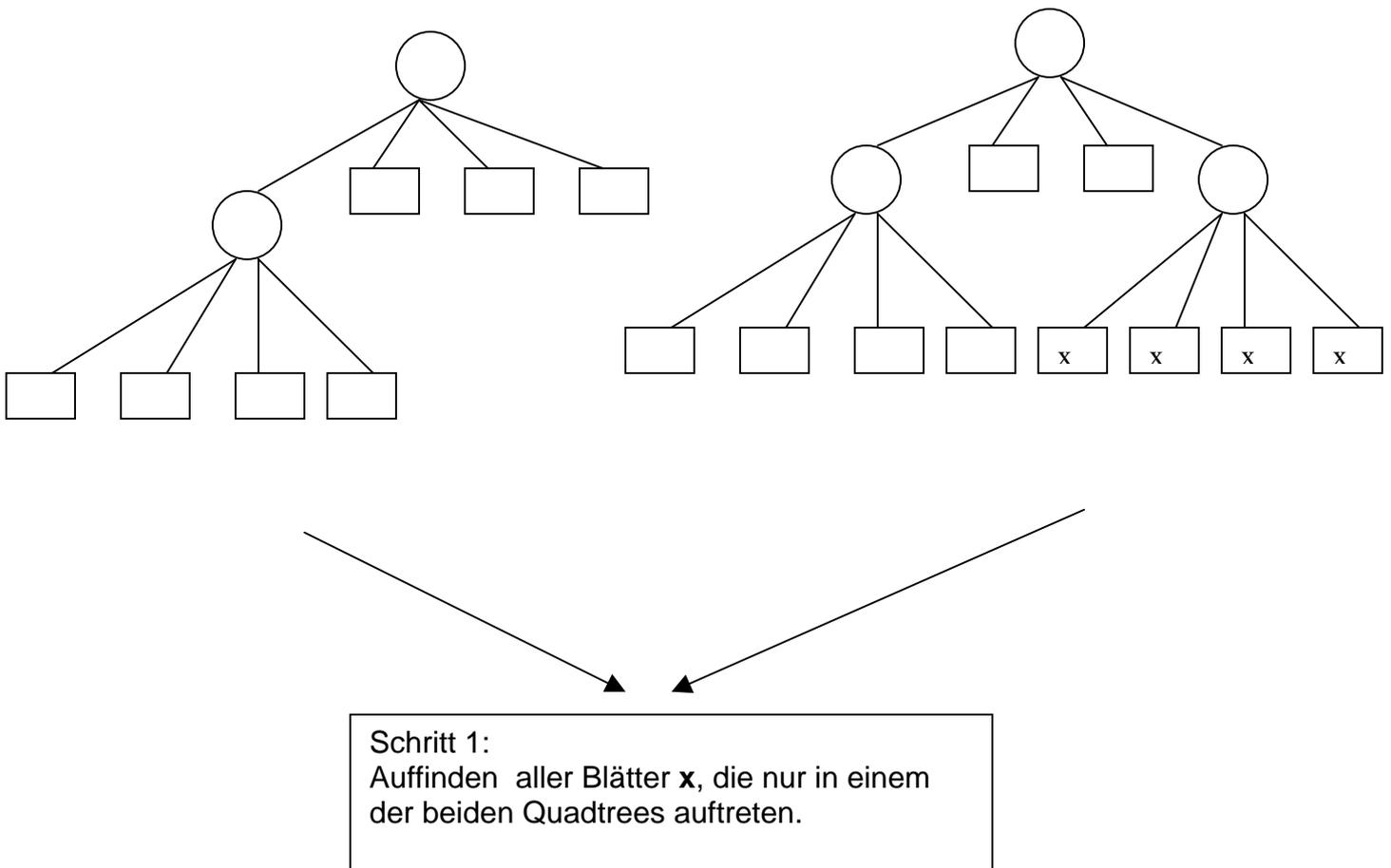
- a.) Wurzel ist Startwert
- b.) Zufällige Auswahl eines Knotens k als Blatt.
- c.) Auswahl der Disparität, welche die Summe der Unterschiedlichkeiten aller Pixel im Knoten k minimiert.
- d.) Zuordnen der gefundenen Disparität zu Knoten k .
- e.) Das Ergebnis sind Disparitätenkarten unterschiedlicher Auflösungsstufen

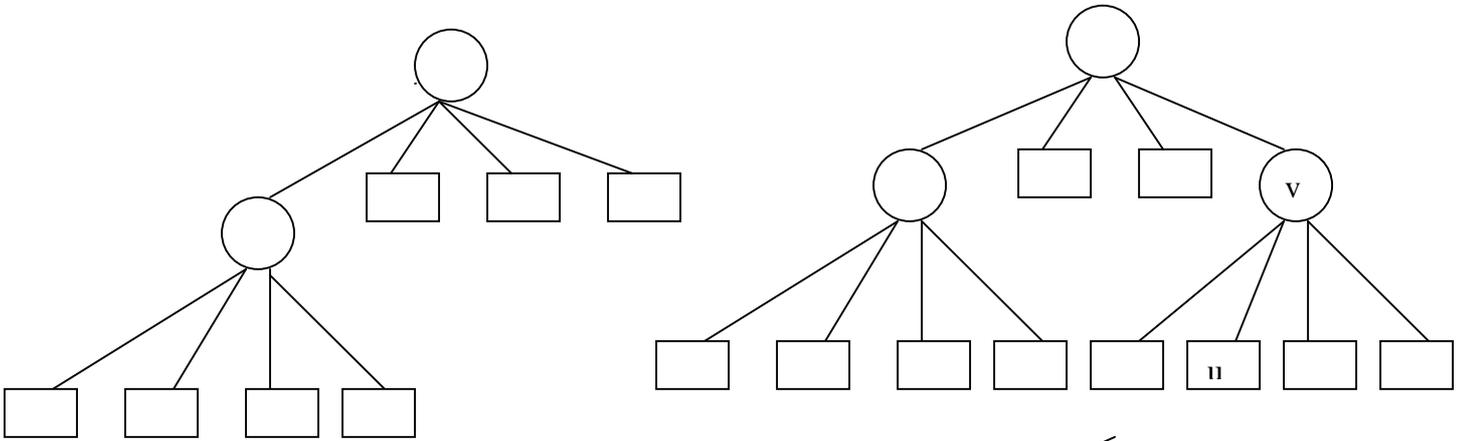
6. Der Crossover Operator

Der Crossover ist ein Genrekombinationsoperator, welcher die Entstehung von Individuen mit neuen Merkmalen ermöglicht.

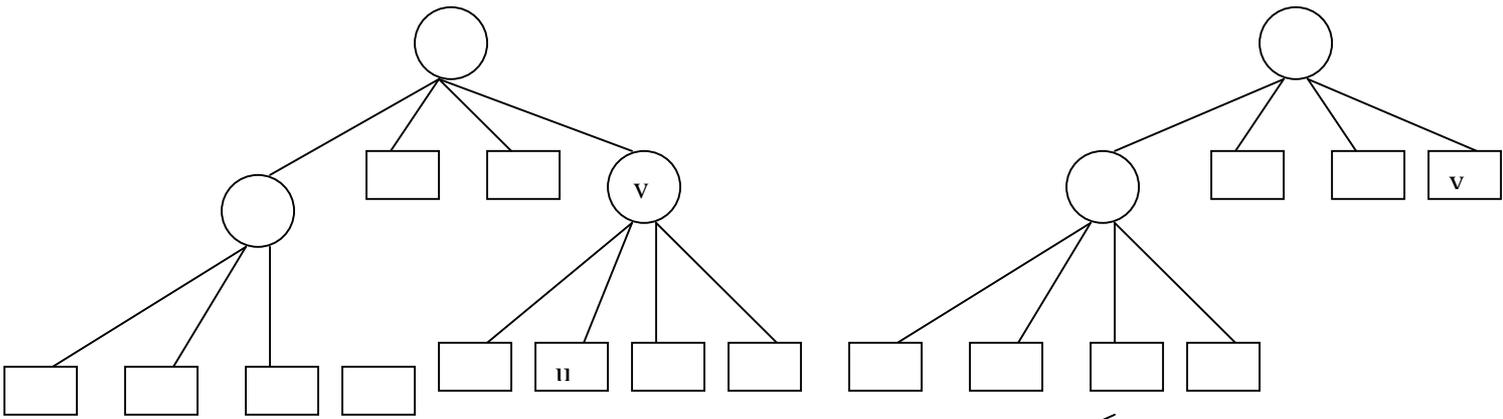
Der Effekt des Crossover-Operators lässt so im Laufe der Zeit typischerweise nach, je mehr die Population zu einer Lösung hinkonvergiert. Dadurch steigt die Wahrscheinlichkeit, dass zwei gleiche oder fast gleiche gekreuzte Genome mindestens ein Kindgenom erzeugen, welches mit einem der beiden Elternteile identisch ist.

In dem hier vorgestellten Algorithmus werden nun zwei Quadrees verglichen. Bei diesem Vergleichsprozess entstehen zwei neue Quadrees.





Schritt 2:
 Zufälliges selektieren eines der gefundenen
 Blätter, den so genannten **Seed Node x**.
 Und Auffinden des Vorgängerknotens v
 (**Cover Node**)



Schritt 3:
 Alle Nachfolger des Cover Nodes an den
 anderen Quadtree anhängen und aus dem
 Original entfernen.(Vertauschung)

7. Der Mutations Operator

Da der Crossover Operator keine Gene erzeugen kann, die in der Initialpopulation nicht enthalten sind benötigt man einen Operator der dieses Problem löst. Dieses ist der Mutationsoperator.

Vergleichbar ist seine Wirkung mit der Wirkung von Strahlung auf das menschliche Erbgut (siehe Tschernobyl und seine Folgen)

In diesem Algorithmus wird dies in drei möglichen Operationen auf den Quadrees ausgedrückt.

Merging (Mischen)

Altering (Veränderung)

Splitting (Aufspaltung)

Merging: bildet eine Mischung (z.B. Durchschnitt) des Blattes k mit seinen Geschwistern

Altering: Änderung des Wertes von Blättern

Splitting: Ein Blatt wird dabei zu einem Knoten mit Blättern

Die Operationen Merging, Altering und Splitting werden nur durchgeführt, wenn dabei die Energie gesenkt wird (-> siehe fitness function). Merging findet zudem nur statt wenn keines der Nachbarblätter (Geschwister) Nachfolger (Kinder) besitzt.

8. Der Minimierungsprozess

Mutation und Crossover werden auf zwei zufällige Quadrees angewendet. Dieser Prozess wiederholt sich so lange bis alle Quadrees (Strings) abgearbeitet sind.

Um die Konvergenz zur gesuchten Disparity Map zu erhöhen wird die Elitist Strategie angewendet. Dabei wird der Quadtree, der die fitness function am besten minimiert ohne Anwendung der Operatoren in die nächste Generation übernommen. Er ersetzt den Quadtree, der die fitness function am schlechtesten erfüllt.

Als Abbruch des Algorithmus wird ein Unterschied von 0.01% zwischen diesen beiden Quadrees gewählt.

9. Parameter free measures

Finden der besten Disparity Map aus den Entstanden anhand des Kriteriums:

$$G(I) = \frac{\sqrt{R}}{10^6 (M \times N)} \times \sum_{i=1}^R \frac{E_i^2}{\sqrt{A_i}}$$

- R: Anzahl der Regionen (Regionen: Pixel gleicher Disparität in der Vierernachbarschaft)
- $M \times N$: Bildgröße
- A_i : Größe einer Region
- E_i : Summe der Quadrate der euklidischen Distanz der Farbvektoren zwischen segmentiertem und Originalbild

10. Beispiele

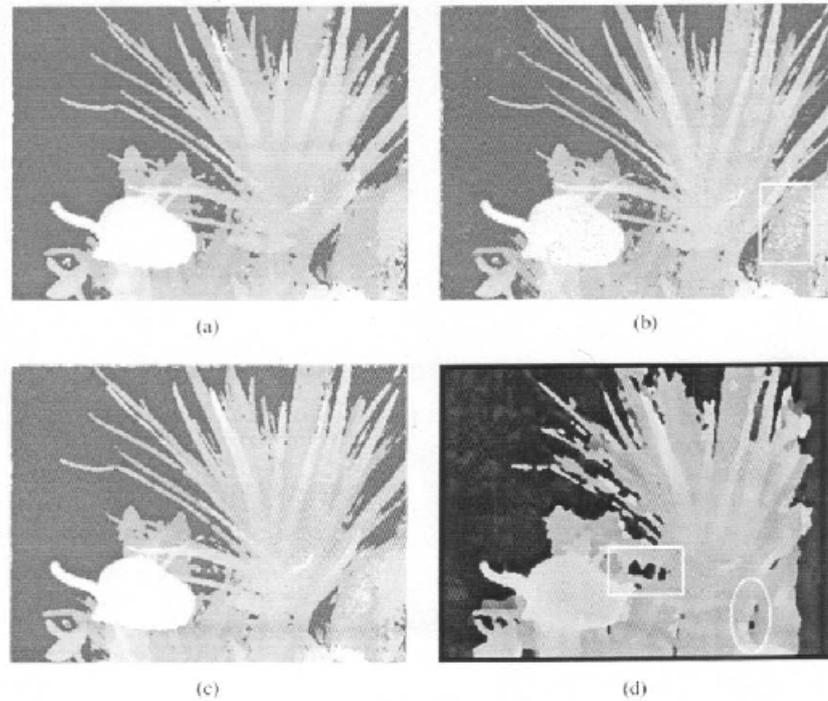


Figure 7. "Plant" multi-view stereo (a) our approach ($\lambda = 1$) (b) SEA (3×3 window) (c) SEA (5×5 window) (d) cooperative algorithm (disparity [12, 51]).

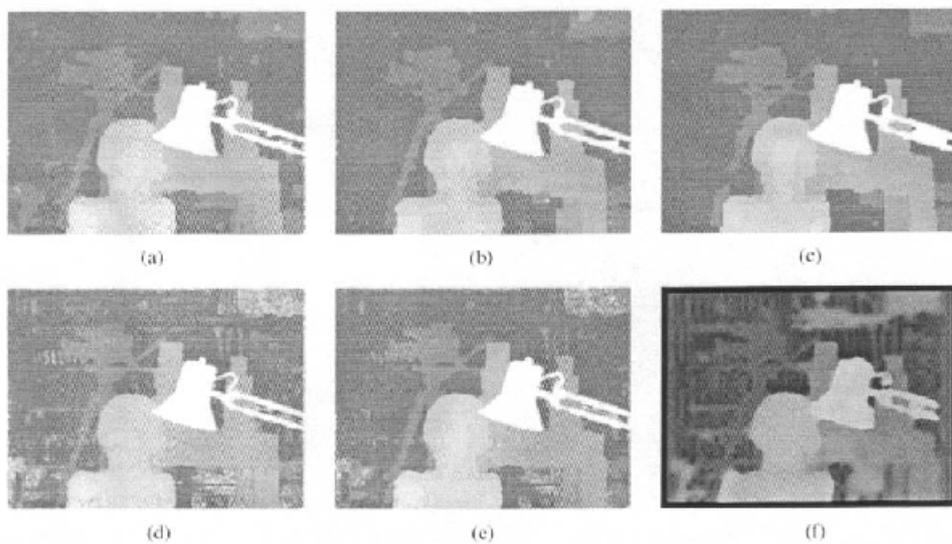


Figure 6. "Head and lamp" multi-view stereo (a) our approach ($\lambda = 1$) (b) our approach ($\lambda = 4$) (c) our approach ($\lambda = 14$) (d) SEA (3×3 window) (e) SEA (5×5 window) (f) cooperative algorithm.

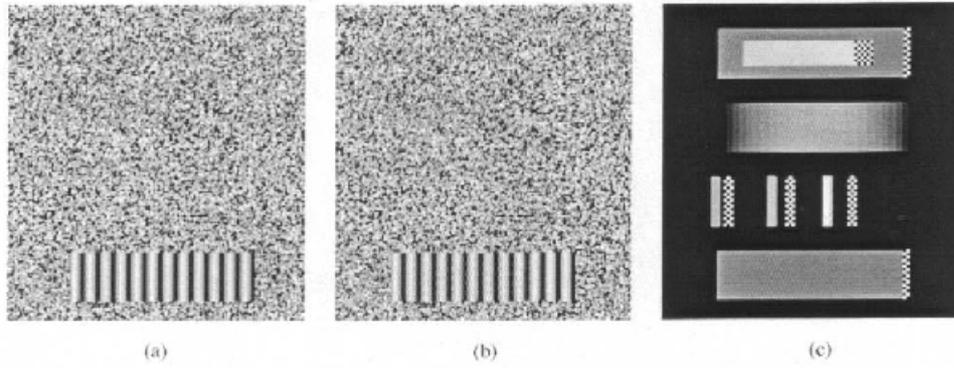


Figure 5. "Random dot stereogram" (a) reference (left) image (b) right image (c) ground truth.

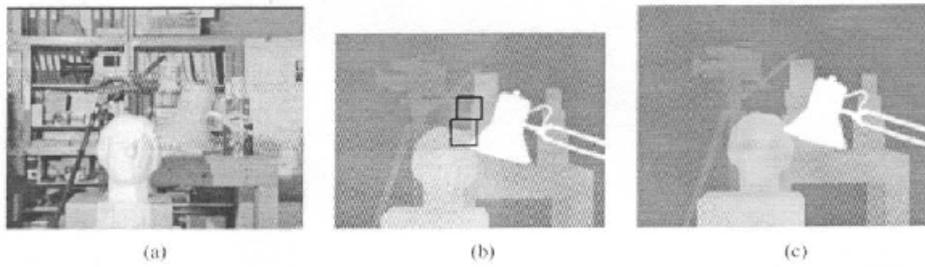


Figure 2. "Head and lamp" multi-view stereo (a) center image (b) ground truth (c) updated ground truth.

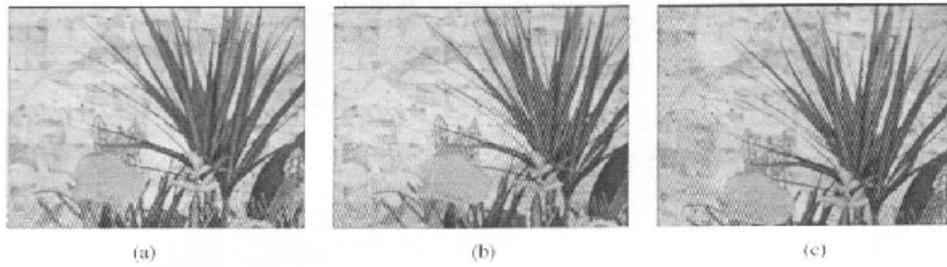


Figure 3. "Plant" multi-view stereo (a) center image (b) left (reference) image (c) top (reference) image.

This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.