

МЕТОДИ ЗА ОТКРИВАНЕ НА ХАРАКТЕРНИ ТОЧКИ И ТЯХНОТО СЪОТВЕТСТВИЕ В СТЕРЕОДВОЙКИ ИЗОБРАЖЕНИЯ

Атанас Николов

FEATURE POINTS DETECTION METHODS AND THEIR CORRESPONDENCE IN STEREO-IMAGES

Atanas Nikolov

***Abstract:** This paper examines different methods for correspondence estimation between stereo-pair images, based on feature points, previously found by corner detector. The accuracy of these methods depend on the chosen window size (centered on every feature point and is necessary for similarity estimation between two compared regions from the images), in regard to the false matches number and the execution time. A software application is developed for the research goals.*

***Key words:** Moravec corner detector, Harris corner detector, Feature based matching, Match measures, Stereo-pair images.*

1. Въведение

Фактът, че светът е съставен от триизмерни обекти, а информацията регистрирана от дадена камера е двумерна, води до необходимостта от разработване на методи за триизмерното интерпретиране на двумерни изображения.

Основният проблем при възстановяване на триизмерната структура (на дадена сцена) от двойка стереоизображения [5, 8] се състои в намирането на двойки съответстващи точки в тези изображения, представляващи проекции на една и съща точка от триизмерната сцена.

Оценката за съответствието на двойка пиксели е основната стъпка на стереокорелацията, където всеки пиксел от двойката е от различно изображение. Но вземането на единични стойности на интензитета има много ограничения [2]. Затова вместо единични пиксели, методите за оценка на съответствие сравняват области (прозорци) от различни изображения.

Целта на представените в статията резултати е да се покаже, че съществува определена стойност на прозореца, която е оптимална по отношение на брой фалшиви съответствия и време за изпълнение. Също така се наблюдава, че зависимостта между размера на прозореца и броя на фалшивите съответствия е обратно пропорционална, докато времето за изпълнение нараства с увеличаване на размера на прозореца.

В следващите секции е описан метода на Харис (който се базира на метода на Моравек) за откриване на характерни точки „ъгли” в изображения [4], както и различни методи за намиране на съответствие между тези характерни точки [2]. За изследване на влиянието на размера на прозореца върху броя на фалшивите съответствия и времето, е разработено и софтуерно приложение.

2. Методи за откриване на характерни точки

В исторически план са известни много детектори на ъгли, като едни от най-разпространените са тези, предложени от: *Moravec (1977)*, Beudet (1978), Kitchen & Rosenfeld (1982), Forstner (1987), *Harris (1988)*, Deriche (1990), Wang & Brady (1992), SUSAN (1997), CSS (1998), Trajkovic & Hedley (1998), Zheng & Wang (1999) [4].

Въпреки че съществуват значителен брой, различни подходи за откриване на характерни точки (ъгли) в полутонови изображения, те могат да се обобщят в три основни направления: методи, основаващи се на ръбове, топологични методи и *автокорелационни методи* [4].

В тази секция ще бъдат разгледани автокорелационните методи за откриване на ъгли, предложени от Моравек и Харис, тъй като те са исторически значими, широко използвани и приложими в реално време.

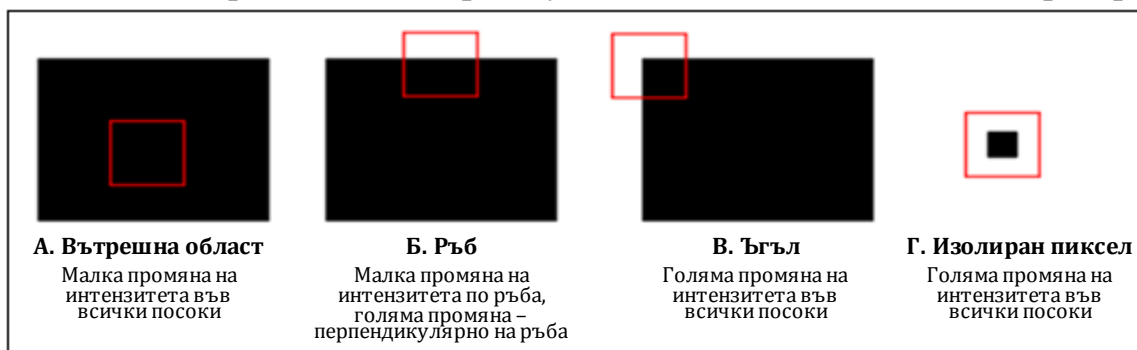
2.1. Оператор на Моравек за откриване на „характерни точки”

Този оператор е бил разработен от Hans P. Moravec през 1977 г. за неговото изследване, включващо навигация на движещо се средство, снабдено с камера, управлявано от компютър, през разнообразна заобикаляща среда [4].

Моравек дефинира, понятието „характерни точки”, като точки, където има голяма промяна на интензитета във всички посоки. Това е случаят на намиране на ъгли, за разлика от ръбове, където има голяма промяна на интензитета само в дадена посока [4].

Моравек предлага измерване на промяната на интензитета чрез поставяне на малък квадратен прозорец (типично 3x3, 5x5 или 7x7 пиксела), центриран около даден пиксел (x, y), и след това преместване на този прозорец с един пиксел във всяка от осемте основни посоки (хоризонтална, вертикална и 4 диагонални). Промяната на интензитета за дадено отместване се изчислява чрез вземане на сумата от квадратните разлики на интензитета на съответстващите пиксели във всеки два прозореца. Тази операция се повтаря за всяка позиция на даден пиксел, на която се присвоява стойност, равна на минималната промяна на интензитета, изчислена в осемте посоки. Характерните точки са локалните максимуми на „характерните стойности”, като резултат от прилагане на праг и потискане на нелокалните максимуми [4].

За да се разбере защо операторът на Моравек е детектор на ъгли, е необходимо да се разгледат четири случая за местоположението на прозореца:



Фиг. 1: Различни случаи на местоположението на прозореца за оператора на Моравек

1) Позиция *A* е вътрешната част на обекта (или на фона), където се предполага, че интензитетът е относително постоянен в прозореца, така че преместването му в коя да е посока ще доведе само до малки промени на интензитета.

2) За прозорец, който се намира върху ръб, както е в позиция *B*, преместването му перпендикулярно на ръба ще доведе до голяма промяна в интензитета, докато преместването му по посока на ръба ще се отрази само в малка промяна на интензитета.

3) За двете позиции *B* и *G*, които съответно отговарят на ъгъл и на изолиран пиксел, ще се получат големи промени в интензитета за всички посоки на преместване на прозореца.

Това показва, че операторът на Моравек наистина е детектор на ъгли, но също той е чувствителен към маркиране на изолирани пиксели като ъгли. Затова при наличие на голяма промяна на интензитета във всички посоки дори вземането на минималната промяна в една от посоките ще бъде голяма и тази точка ще се определи като ъгъл.

Математическото описание на горното обяснение, може да бъде изразено формално чрез следния алгоритъм [4]:

Означаваме интензитета на пиксел с координати (x, y) с $I(x, y)$.

Вход: полутоново изображение, размер на прозореца, праг T .

Изход: карта, показваща позицията на всеки намерен ъгъл.

1. За всеки пиксел (x, y) в изображението се изчислява изменението на интензитета в резултат от отместването (u, v) на прозореца в осем посоки:

$$V_{u,v}(x, y) = \sum_{\forall a, b \text{ в прозореца}} w(a, b) [I(x + u + a, y + v + b) - I(x + a, y + b)]^2,$$

където $w(a, b)$ е квадратен прозорец с тегла равни на 1, а разглежданите отмествания (u, v) на прозореца са: $\{(1,0), (1,1), (0,1), (-1,1), (-1,0), (-1,-1), (0,-1), (1,-1)\}$.

2. Съставя се карта с ъгли чрез изчисляване на измерването $C(x, y)$, което определя степеня, с която даден пиксел (x, y) се счита за ъгъл:

$$C(x, y) = \min (V_{u,v}(x, y))$$

3. Към картата с ъгли се прилага праг T , като се присвоява стойност нула на всички стойности на $C(x, y)$, които са под прага.

4. Извършва се потискане на немаксималните стойности, за да се намерят локалните максимуми.

Всички ненулеви точки, останали в картата с ъгли, са търсените ъгли.

Като обобщение може да се каже, че детекторът на Моравек се състои в следното: търсене на локален максимум в $\min \{V\}$ над някаква прагова стойност.

Ограничения на оператора на Моравек [4]:

1. Анизотропен резултат - дължи се на факта, че изменението на интензитета се изчислява само за дискретно множество от отмествания - в осем посоки. Затова операторът е чувствителен към ротация;

2. Чувствителност към шум - дължи се на това, че прозорецът, използван от Моравек, е квадратен и двоичен.

3. Голяма чувствителност към ръбове - поради несъвършенства в ръбовете, дължащи се на шум или дискретизация на интензитета, като в резултат на това локалният минимум от разликите на интензитетите по всички посоки на отместване ще бъде голям. Ръбове с определена ориентация също твърде лесно се маркират като ъгли, понеже разликите в интензитета се изчисляват само в осем посоки.

2.2. Оператор на Харис за откриване на „характерни точки“

Операторът на Моравек е от голямо значение, понеже той е вдъхновил широко използвания оператор на Харис, който, както в трудовете на Моравек, въвежда концепцията за „характерни точки.“ Много от слабостите на оператора на Моравек са разрешени от оператора на Харис. Той се различава от оператора на Моравек по начина, по който се изчислява локалната автокорелация, което позволява да бъде измерена промяната на интензитета по всички посоки [3]. Харис изчислява измерване на локалната автокорелация, използвайки производни от първи ред, което е извършено чрез аналитично разширяване на оператора на Моравек чрез ред на Тейлър на функция на две променливи [4].

Нека $f(x, y)$ да бъде функция на две променливи и се предполага, че има непрекъснати частни производни от всички редове в съседство (x_0, y_0) . Тогава е в сила следната редица:

$$f(x, y) = f(x_0, y_0) + f_x(x_0, y_0)(x - x_0) + f_y(x_0, y_0)(y - y_0) + H(x, y),$$

където членовете от по-висок ред H се изразяват чрез:

$$H(x, y) = \frac{1}{2!} (f_{xx}(x_0, y_0)(x - x_0)^2 + 2f_{xy}(x_0, y_0)(x - x_0)(y - y_0) + f_{yy}(x_0, y_0)(y - y_0)^2) + \dots$$

Сега, нека $f(x + u, y + v)$ да бъде функцията на две променливи и да има непрекъснати частни производни в съседство (x, y) . Тогава се получава следната редица [1]:

$$f(x + u, y + v) = f(x, y) + u f_x(x, y) + v f_y(x, y) +$$

първи частни производни

$$\frac{1}{2!} [u^2 f_{xx}(x, y) + 2uv f_{xy}(x, y) + v^2 f_{yy}(x, y)] + \dots$$

втори частни производни

Следователно, приближението на първата производна на функцията $f(x + u, y + v)$ е:

$$f(x + u, y + v) \approx f(x, y) + u f_x(x, y) + v f_y(x, y)$$

Имайки предвид горното приближение, операторът на Моравек може да се запише по следния начин [1]:

$$V(u, v) = \sum_{\forall x, y \text{ в прозореца}} w(x, y) [I(x + u, y + v) - I(x, y)]^2 \approx$$

$$\sum_{x, y} w(x, y) [I(x, y) + u I_x + v I_y - I(x, y)]^2 = \sum_{x, y} w(x, y) [u I_x + v I_y]^2 =$$

$$\sum_{x, y} w(x, y) (u^2 I_x^2 + 2uv I_x I_y + v^2 I_y^2) = Au^2 + 2Cuv + Bv^2 ,$$

където

$$A = I_x^2 \otimes w = \left(\frac{dI}{dx} \right)^2 \otimes w, \quad B = I_y^2 \otimes w = \left(\frac{dI}{dy} \right)^2 \otimes w, \quad C = I_x I_y \otimes w = \left(\frac{dI}{dx} \frac{dI}{dy} \right) \otimes w$$

\otimes е оператор за конволюция,

$w(x, y)$ е съответното тегло на Гаусовия прозорец на позиция (x, y) ,

$$I_x = I \otimes (-1, 0, 1) \approx \frac{dI}{dx} \quad \text{и} \quad I_y = I \otimes (-1, 0, 1)^T \approx \frac{dI}{dy}$$

Харис и Стийфън забелязват, че това може да се запише като матрично уравнение [4]:

$$V(u, v) = Au^2 + 2Cuv + Bv^2 = [u \quad v] M \begin{bmatrix} u \\ v \end{bmatrix} ,$$

където

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} = \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

Матрицата M съдържа всички диференциални оператори, описващи геометрията на повърхността на изображението за дадена точка (x, y) . Собствените стойности на M са пропорционални на кривините на повърхността на изображението и формират нечувствителна на ротация характеристика на M .

Както при оператора на Моравек, тук също трябва да се разгледат четири случая за позицията на прозореца (*Фиг. 1*):

1) Позиция A е вътрешна част на обект (или на фон), където се предполага, че интензитетът на изображението е относително непроменлив в прозореца. Понеже съществува малка кривина на повърхността в този прозорец, двете собствени стойности на M ще бъдат относително малки.

2) За локалния прозорец, намиращ се върху ръба, както е в позиция B , има значително изкривяване, перпендикулярно на ръба, и много малка кривина по протежение на ръба, така че едната собствена стойност на M ще бъде голяма, а другата - малка.

3) Двете позиции B и Γ , които съответстват на ъгъл и изолиран пиксел, ще имат значително изкривяване в две посоки, така че двете собствени стойности на M ще бъдат големи.

Нека да се означат характеристичните корени на M с λ_1 и λ_2 . Според горния анализ, равнината описана от λ_1 и λ_2 , може да се раздели на три различни области, както е показано на *Фиг. 2*.

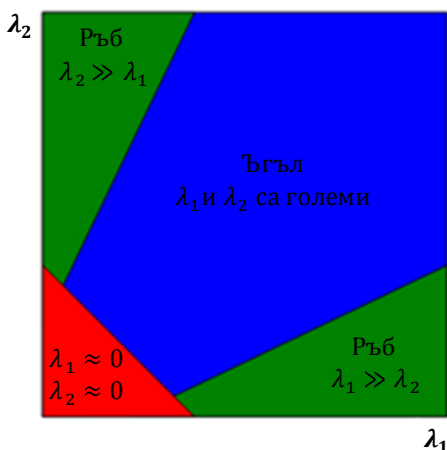
Харис и Стифън предлагат следното измерване на степента, с която се определя дали даден пиксел е ъгъл [3, 4]:

$$(1) \quad C(x, y) = \det(M) - k(\text{trace}(M))^2,$$

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix},$$

където

$$\det(M) = \lambda_1 \lambda_2 = AB - C^2, \quad \text{trace}(M) = \lambda_1 + \lambda_2 = A + B, \quad k = 0.04 \div 0.06$$



Стойността на k е избрана в този диапазон емпирично, понеже дава най-добри резултати. Според формулата за $C(x, y)$ и стойностите на λ_1 и λ_2 за трите области от *Фиг. 2.*, стойността на $C(x, y)$ ще бъде отрицателна за ръбове и положителна на ъгли и вътрешни части на обекти. Затова трябва да се приложи прагова стойност, за да се различат ъглите от вътрешните точки. Въпреки че стойността от измерването $C(x, y)$ на вътрешните точки е много малка,

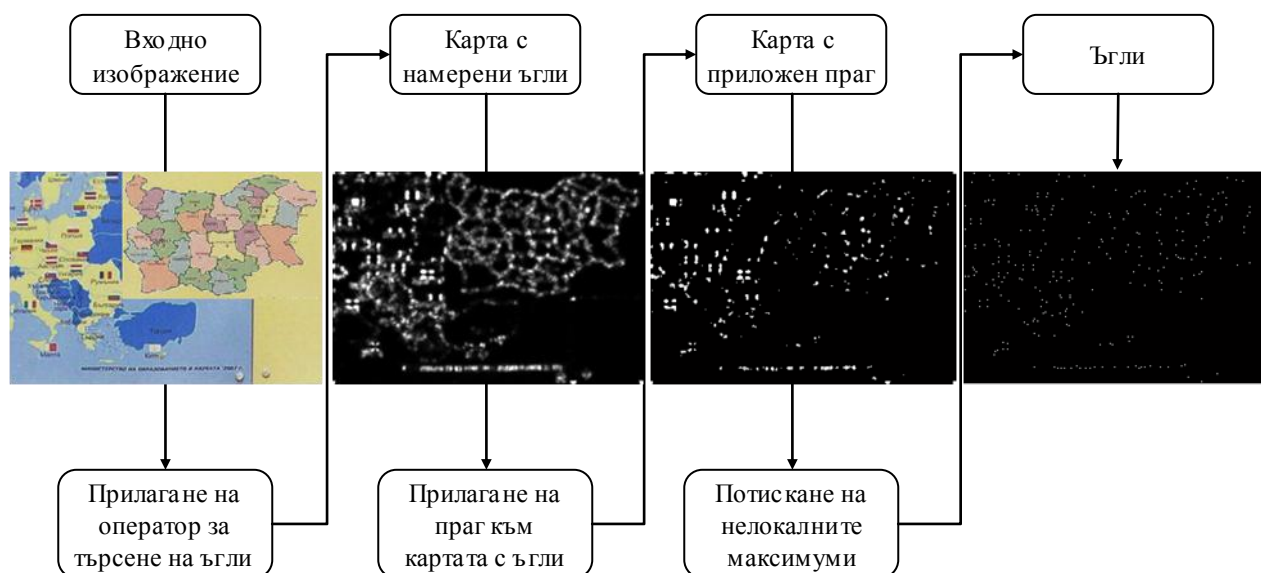
Фиг. 2: Разделяне на равнината, описана от собствените стойности на M , на различни области за класификация на пикселите от изображението.

прагът трябва да бъде достатъчно висок, за да се предотврати откриването на фалшиви ъгли, които може да имат относително голяма стойност, дължаща се на шум. Когато стойностите на λ_1 и λ_2 са големи, тогава и стойността от измерването $C(x, y)$ е голяма, което означава, че ъглите са локалните максимуми в картата от ъгли, след прилагането на определена прагова стойност [4].

Използването на емпиричния параметър k във (формула 1) и стойността, която трябва да му се присвои е проблем за някои хора. Alison Noble [6] представя алтернативно измерване за пикселите на степента им на подобие с ъгли чрез следното отношение:

$$(2) \quad C(x, y) = \det(M) / \text{trace}(M)$$

В разработеното софтуерно приложение се използва предложената от Alison Noble оценка за намиране на ъгли (формула 2), приложена за всеки пиксел от изображението според следната схема за откриване на ъгли:



Фиг. 3: Графично представяне на алгоритъма за откриване на ъгли

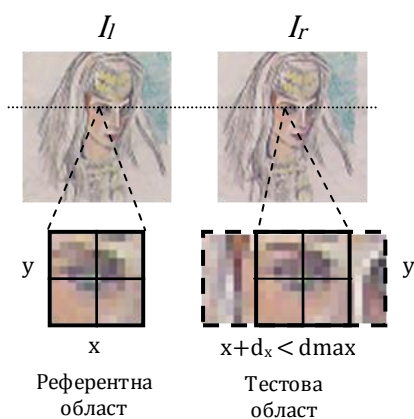
3. Методи за намиране на съответствия между характерни точки

В компютърното зрение, съответствието (степената на подобие) между две групи от данни принадлежи към фундаменталните методи за обработка на изображения. Две точки от изображенията са съответстващи (съвпадат), ако те са резултат от проекцията на една и съща триизмерна точка от сцената [8]

Трудностите пред алгоритмите за съответствие се дължат на съществуването на двусмислени (неопределени) съответствия в резултат на области с еднаква текстура, както и съмнителни стойности на интензитета в резултат на въведен шум от процеса на създаване на изображението [7]. Друг проблем са частично закритите точки, т.е. точки, видими само от едната камера, за които не може да се търси съответствие. Освен това, основната причина, която прави трудно съответствието, е, че количеството информация, налично за

единичен пиксел (т.е. интензитет, който обикновено е повлиян от шум), не е достатъчно за намиране на недвусмислено съответствие. Понеже съответствието между единични пиксели е нестабилен процес, е необходимо да се разгледа малко локално съседство около всеки пиксел, за да се намали неопределеността. Също така често само за области с голямо количество информация като интензивни ръбове или ъгли може недвусмислено да се определи съответствието [8].

За да се намери стойността на отместването (disparity) за всяка характерна точка (x, y) в референтното изображение, трябва да се намери най-доброто съответстващо местоположение от сравняваните области в другото изображение. Табл. 1 съдържа списък на основните методи [2] за измерване на съответствие на дадени региони от изображения, които оперират директно върху пиксели. Използвайки такива метрики, можем да разгледаме локално съседство N на пиксел x, y в лявото изображение I_l и да търсим за съответстващо местоположение в дясното изображение I_r от стереодвойката, използвайки същото по размер съседство. Този процес на локално търсене се



Фиг. 4: Търсене на съответстващи области в различни изобр.

които са отместени само по хоризонтала, получени от канонична система за стереозрение [5, 8].

прилага, за да се намери локално съседство в I_r , което да е най-подобно на референтното съседство, разположено в лявото изображение I_l . Този процес се повтаря за всяка характерна точка (ъгъл) в I_l , за да се намери най-подобното и съседство в I_r . По този начин може да се получи стойност на отместването за всички характерни точки x, y в I_l . На Фиг. 4. е показано обикновено 1D търсене, при което дадена референтна област от лявото изображение се сравнява с всички възможни области, намиращи се на определено разстояние върху епиполярната линия от дясното изображение [2]. Това се отнася за изображения,

Таблица 1: Методи за измерване на съответствието между различни области от изображения

Име	Метод
Sum of Absolute Differences	$SAD = \sum_{(i,j) \in U} I_1(x+i, y+j) - I_2(x+d_x+i, y+d_y+j) $
Zero mean Sum of Absolute Differences	$ZSAD = \sum_{(i,j) \in U} [I_1(x+i, y+j) - \overline{I_1(x,y)}] - [I_2(x+d_x+i, y+d_y+j) - \overline{I_2(x+d_x, y+d_y)}] $

Sum of Squared Differences	$SSD = \sum_{(i,j) \in U} [I_1(x+i, y+j) - I_2(x+d_x+i, y+d_y+j)]^2$
Zero mean Sum of Squared Differences	$ZSSD = \sum_{(i,j) \in U} [(I_1(x+i, y+j) - \overline{I_1(x,y)}) - (I_2(x+d_x+i, y+d_y+j) - \overline{I_2(x+d_x, y+d_y)})]^2$
Covariance-Variance	$CV = \frac{\sum_{(i,j) \in U} [I_1(x+i, y+j) - \overline{I_1(x,y)}] \cdot [I_2(x+d_x+i, y+d_y+j) - \overline{I_2(x+d_x, y+d_y)}]}{\sqrt{\sum_{(i,j) \in U} [I_1(x+i, y+j) - \overline{I_1(x,y)}]^2 \cdot \sum_{(i,j) \in U} [I_2(x+d_x+i, y+d_y+j) - \overline{I_2(x+d_x, y+d_y)}]^2}}$
Normalized Cross-Correlation	$NCC = \frac{\sum_{(i,j) \in U} I_1(x+i, y+j) \cdot I_2(x+d_x+i, y+d_y+j)}{\sqrt{\sum_{(i,j) \in U} I_1(x+i, y+j)^2 \cdot \sum_{(i,j) \in U} I_2(x+d_x+i, y+d_y+j)^2}}$

Във всички дефиниции се приема, че се сравняват два съвместими региона I_1 и I_2 . Те могат да принадлежат на едно и също или на различни изображения. Регионът I_1 е построен около отправна точка (x, y) , изразена в нейните локални координати, а I_2 е построен около точката $(x + d_x, y + d_y)$ в нейното локално координатно пространство. Регионите, за които се търси съответствие, са дефинирани чрез множеството U , определено от техните отправни точки, т.е (x, y) и $(x + d_x, y + d_y)$ съответно. Също се приема, че всички индекси, дефинирани в U , попадат в обхвата на валидни местоположения на пикселите за I_1 и I_2 [2].

Таблица 2: Описание на означенията, използвани в Таблица 1

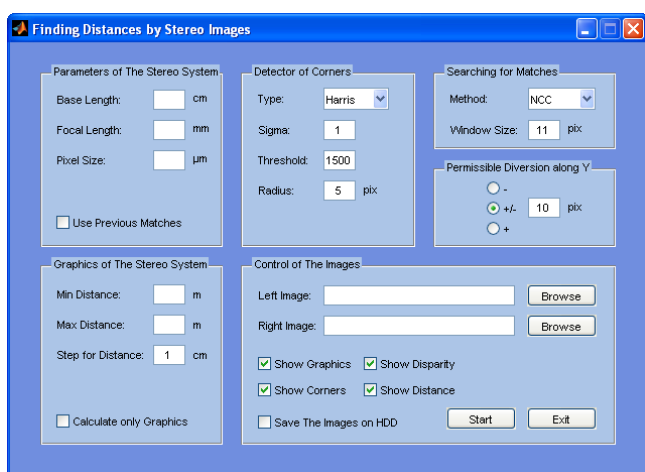
Означение от Таблица 1	Описание
$U = U(x, y)$	Множество от точки (координати на точки), разположени около точка с координати (x, y) в изображението.
$I_k(x, y)$	Стойност на интензитета на точка с локални координати (x, y) в k -тото изображение.
$\overline{I_k(x, y)} = \frac{1}{N} \sum_{(i,j) \in U} I_k(x+i, y+j)$ където N = броя на елементите в U	Средна стойност на интензитета в съседство U около точка с локални координати (x, y) в k -тото изображение. N обозначава броя на точките, които се използват в изчислението, т.е. броят на различните двойки (i, j) , които обозначават относителното отместване около (x, y) . Те могат да бъдат положителни или отрицателни, но трябва да принадлежат на област от k -тото изображение.
d_x, d_y	Параметри, които обозначават относителното хоризонтал-но и вертикално отместване на два блока (области U) от изображенията, които се сравняват.

Изборът на прозореца U дори е по-обременително, отколкото изборът на метод за търсене на съответствия. Не съществуват стриктни правила за определяне на U и изборът зависи от приложението и съдържанието на изображението. Ако се избере твърде малък прозорец, може да не се прихванат достатъчно елементи от изображението и да се увеличи чувствителността към шум, т.е. много фалшиви съответствия. Твърде голям прозорец, прави съответствията по-малко чувствителни към шум, но времето за изпълнение нараства значително. Представените резултати в следващата секция са ориентирани основно към избор на оптимален прозорец за постигане на най-добри резултати за броя на фалшивите съответствия и времето за изпълнение.

4. Експериментални резултати

Целта на проведените експерименти е да се изследва точността на различни методи за намиране на съответствия (Табл. 1) в зависимост от размера на избрания прозорец (центриран около всяка характерна точка), необходим за оценка на подобие между две сравнявани области от изображенията.

Експериментите са извършени със стереодвойка изображения, заснети с цифров фотоапарат Kodak C743, хоризонтално транслиран на определено базово разстояние от първоначалната си позиция, с цел имитиране на канонична система за стереозрение [5, 8]. По този начин търсенето за съответствия се ограничава само до $\sim 1D$ търсене (Фиг. 4) по хоризонтални линии (с определено допустимо отклонение по вертикала) и максимално хоризонтално отместване, което отговаря на минималното разстояние до обекти, тъй като дължината на отместването на двойка съответстващи точки и дълбочината до тези точки е обратно пропорционална (Фиг. 6 в, г).



Фиг. 5: Интерфейс на софт. приложение

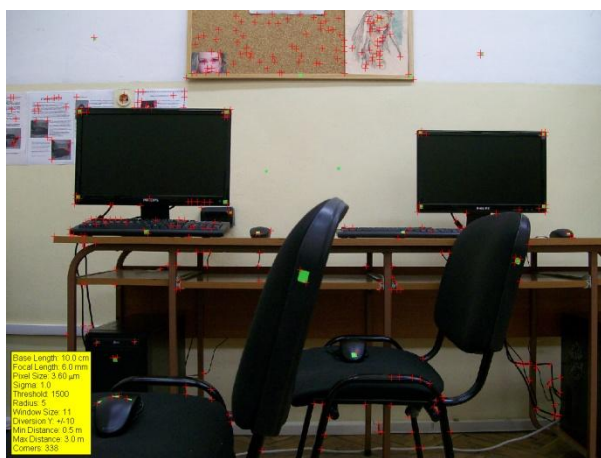
Разработено е софтуерно приложение (Фиг. 5) в програмната среда Matlab v7.11, чрез което се демонстрира откриването на характерни точки в дадена стереодвойка изображения и намирането на тяхното съответствие, след което е лесно да се намери и дълбочината до точка от съответния обект от тримерната сцена [5, 8].

Параметрите, необходими на софтуерното приложение за целите на изследването на различните методи за

оценка на съответствие и различен размер на прозореца, необходим за тази оценка, са: 1) размер на изображенията 1536 x 1152 пиксела, на които отговаря размер на пиксела от 3.6 μm ; 2) базово разстояние между „лявата и дясната“ камера - 10cm; 3) фокусно разстояние - 6 mm; 4) необходимите на детектора на ъгли праг, равен на 1500 и радиус за стъпката „потискане на нелокалните

максимуми”, равен на 5; 5) минимално и максимално разстояние, за които да се търсят съответствията между откритите „ъгли”, равни на 0.5m и 3.0m; 6) допустимо максимално отклонение по вертикала, равно на +/- 10 пиксела, играещо роля на филтър. Този филтър освен, че ограничава съответствията до допустимото им отклонение по вертикала, също така ограничава и хоризонталната дължината между две съответстващи точки, да бъде в определени граници, зависещи от минималното и максимално разстояние за което да се търсят съответствията (Фиг. 6 г).

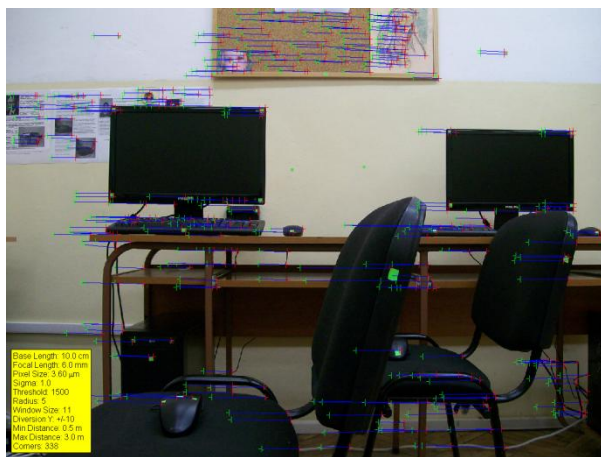
На Фиг. 6 са показани резултати от работата на приложението, свързани с намирането на „ъгли” (а, б) според представения алгоритъм от Фиг. 3 и намирането на съответствия (в) между тези точки според принципа от Фиг. 4.



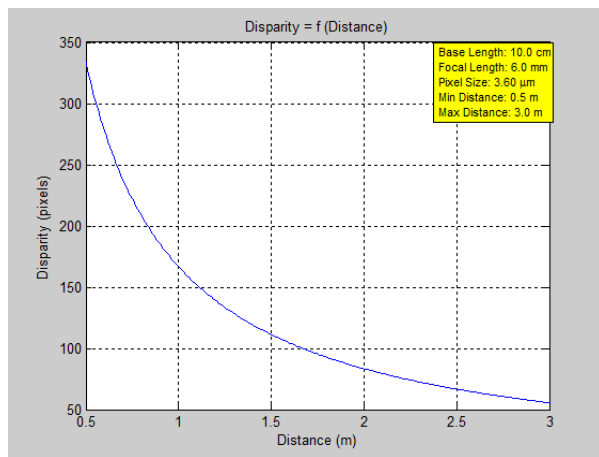
а) ляво изображение с намерени ъгли



б) дясно изображение с намерени ъгли



в) изображение с намерени съответствия



г) зависимост м/у дълбочина и отместване

Фиг. 6: Намерени „ъгли” и съответствия за произволна стереодвойка изображения

В Табл. 3 и 4 са представени резултати от проведени експерименти с различни методи (Табл. 1) за намиране на съответствия между области (прозорци) от две изображения. Експериментите са извършени на компютър (лаптоп) ASUS X51L-AP123L със следните характеристики - процесор: Intel Core2Duo 2GHz, оперативна памет: 2GB DDR2 667MHz, вградена видеокарта: Intel GMA X3100. Изследвано е влиянието на размера на квадратен прозорец (3,5,7,9,11,13,23,...,103,133,173,203 пиксела за страна) върху броя на

намерените фалшиви съответствия, както и влиянието му върху времето за изпълнение на даден метод, при размер на изображенията 1536 x 1152 пиксела.

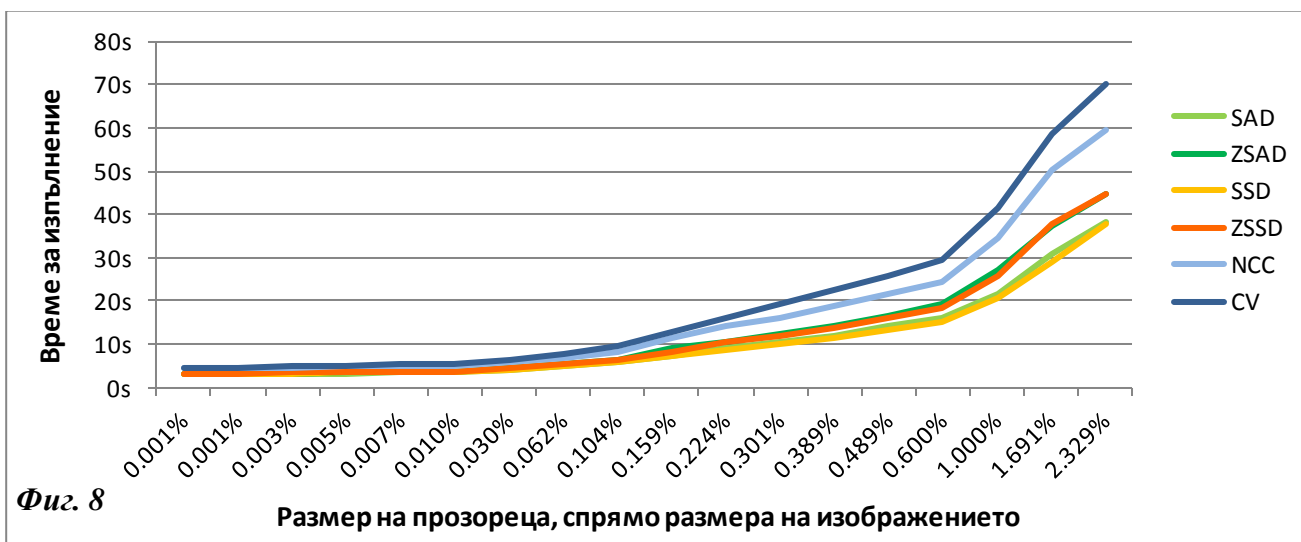
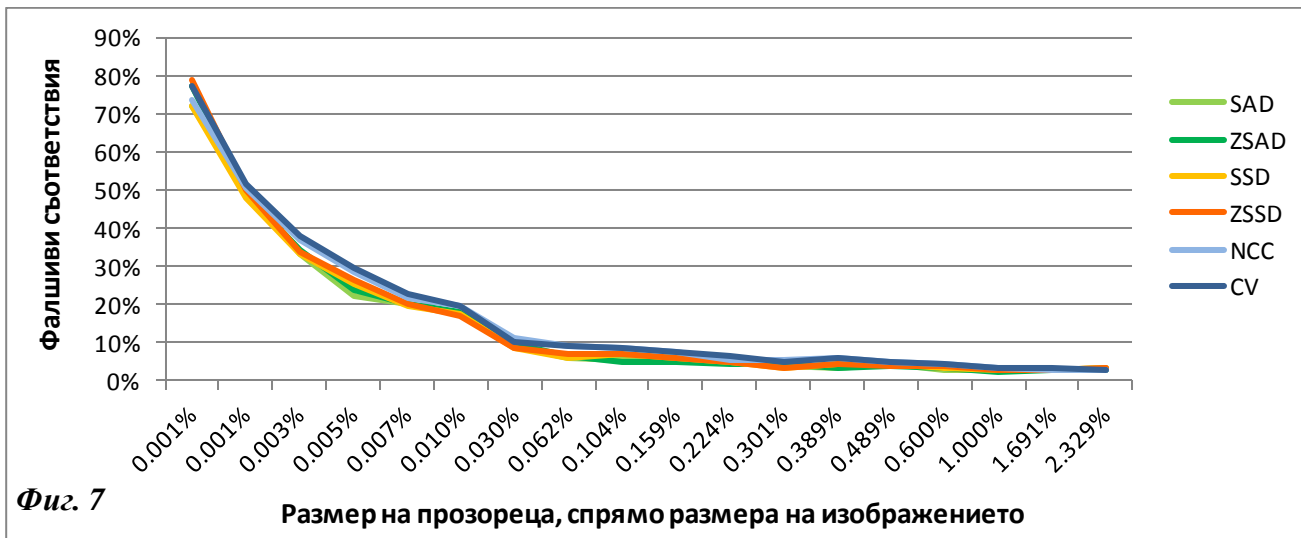
На *Фиг. 7* и *8* са представени графики за: 1) фалшивите съответствия като функция на размера на прозореца, която е обратно пропорционална; 2) времето като функция на размера на прозореца, където зависимостта е правопропорц.

Таблица 3: Резултати за SAD, ZSAD и SSD методите за оценка на съответствие м/у области

размер на прозорец	SAD				ZSAD				SSD			
	време [sec]	намерени съответст.	филтрирани съответст.	грешка [ф/н]	време [sec]	намерени съответст.	филтрирани съответст.	грешка [ф/н]	време [sec]	намерени съответст.	филтрирани съответст.	грешка [ф/н]
0.0005%	2.95	339	95	71.98%	3.13	353	80	77.34%	3.04	337	94	72.11%
0.0014%	3.02	341	170	50.15%	3.21	327	167	48.93%	3.08	332	173	47.89%
0.003%	3.15	363	243	33.06%	3.27	364	239	34.34%	3.17	360	240	33.33%
0.005%	3.37	398	310	22.11%	3.62	391	297	24.04%	3.41	388	289	25.52%
0.007%	3.47	438	349	20.32%	3.64	422	337	20.14%	3.53	418	336	19.62%
0.010%	3.66	461	381	17.35%	3.80	453	371	18.10%	3.62	439	363	17.31%
0.030%	4.17	495	449	9.29%	4.48	492	444	9.76%	4.14	466	426	8.58%
0.062%	5.17	484	454	6.20%	5.30	486	454	6.58%	4.78	454	426	6.17%
0.104%	5.79	487	461	5.34%	6.51	484	459	5.17%	6.03	451	420	6.87%
0.159%	7.36	490	459	6.33%	9.05	484	459	5.17%	7.17	466	438	6.01%
0.224%	9.16	489	464	5.11%	10.52	484	462	4.55%	8.88	449	427	4.90%
0.301%	10.68	476	458	3.78%	12.26	474	454	4.22%	10.28	442	428	3.17%
0.389%	12.10	481	465	3.33%	14.25	482	465	3.53%	11.57	453	433	4.42%
0.489%	14.01	493	472	4.26%	16.43	479	460	3.97%	13.26	460	440	4.35%
0.600%	15.94	489	474	3.07%	19.21	480	463	3.54%	15.25	462	446	3.46%
1.000%	21.82	487	474	2.67%	27.16	485	473	2.47%	20.82	483	470	2.69%
1.691%	30.86	448	435	2.90%	37.22	446	433	2.91%	28.77	433	421	2.77%
2.329%	38.23	423	408	3.55%	44.60	419	406	3.10%	37.77	413	400	3.15%

Таблица 4: Резултати за ZSSD, NCC и CV методите за оценка на съответствие м/у области

размер на прозорец	ZSSD				NCC				CV			
	време [sec]	намерени съответст.	филтрирани съответст.	грешка [ф/н]	време [sec]	намерени съответст.	филтрирани съответст.	грешка [ф/н]	време [sec]	намерени съответст.	филтрирани съответст.	грешка [ф/н]
0.0005%	3.30	349	75	78.51%	4.39	367	98	73.30%	4.65	369	84	77.24%
0.0014%	3.26	329	164	50.15%	4.48	370	184	50.27%	4.70	375	182	51.47%
0.003%	3.42	357	237	33.61%	4.62	396	249	37.12%	4.82	398	247	37.94%
0.005%	3.59	384	282	26.56%	4.83	415	297	28.43%	5.09	411	290	29.44%
0.007%	3.72	418	333	20.33%	5.10	433	338	21.94%	5.24	436	336	22.94%
0.010%	3.81	435	361	17.01%	5.08	454	365	19.60%	5.39	451	362	19.73%
0.030%	4.62	465	424	8.82%	5.79	490	436	11.02%	6.40	478	430	10.04%
0.062%	5.39	456	424	7.02%	6.83	504	458	9.13%	7.94	507	461	9.07%
0.104%	6.34	451	420	6.87%	8.16	494	452	8.50%	9.39	489	447	8.59%
0.159%	8.30	458	430	6.11%	11.29	506	469	7.31%	12.63	503	465	7.55%
0.224%	10.33	447	425	4.92%	14.10	494	466	5.67%	16.15	489	458	6.34%
0.301%	12.00	439	424	3.42%	16.22	485	459	5.36%	19.19	485	460	5.15%
0.389%	13.90	450	430	4.44%	18.68	484	456	5.79%	22.39	487	458	5.95%
0.489%	16.04	457	438	4.16%	21.59	489	464	5.11%	25.56	489	464	5.11%
0.600%	18.28	460	443	3.70%	24.38	494	471	4.66%	29.27	492	470	4.47%
1.000%	25.70	478	464	2.93%	34.35	494	478	3.24%	41.31	488	472	3.28%
1.691%	37.95	430	417	3.02%	50.30	447	434	2.91%	58.45	449	435	3.12%
2.329%	44.57	410	396	3.41%	59.69	426	413	3.05%	70.29	426	414	2.82%



Изводи:

- Всички изследвани методи дават задоволително добри резултати, за които от най-голямо значение е изборът на оптимален размер на прозореца, по отношение на броя на фалшивите съответствия и времето за изпълнение.
- Най-малък процент фалшиви съответствия (3%) за всички методи се получава, когато размерът на прозореца е 1% от размера на изображението.
- За този размер на прозореца, времето за изпълнение на методите е между 20 и 40 секунди, което означава, че такъв размер на прозореца не е оптимален.
- Един добър размер на прозореца за всички изследвани методи, при който фалшивите съответствия са под 10%, а времето за изпълнение е под 6.5 сек., е 0.03% от размера на изображението.
- Най-бързите от изследваните методи са SAD и SSD, следват ZSAD и ZSSD, а най-бавни са NCC и CV за всеки размер на прозореца.
- Графиките на всички методи почти съвпадат по отношение на броя на фалшивите съответствия за всички размери на изследваните прозорци.

5. Заключение и бъдеща работа

В настоящата статия се изследват различни методи за намиране на съответствия между области от две изображения, целта на което е да се определи оптималният размер на прозореца (областта за сравнение), с който да се получат относително малък брой фалшиви съответствия, както и относително кратко време за изпълнение на методите.

Бъдещото развитие може да се очертае в следните насоки: разглеждане на всяка една точка от изображенията като характерна и намирането на съответствието ѝ в другото изображение; определяне на тримерните координати (X, Y, Z) на произволна точка, като резултат от намерените съответствия, където алгоритъмът за определяне на дълбочина (Z), да не изисква задължително използването на принципа на канонична стереосистема [5, 8]; създаване на метод за 3D моделиране и разпознаване на обекти.

Благодарности

Тази разработка е подкрепена финансово от Проект No: BG 051 PO 001-3.3.04/13, "Подкрепа на творческото развитие на докторанти, пост-докторанти и млади учени в областта на компютърните науки", финансиран от ЕВРОПЕЙСКИ СОЦИАЛЕН ФОНД, Оперативна програма "Развитие на човешките ресурси" 2007-2013.

Литература

[1] *Collins, R.* Lecture Notes on Computer Vision, CSE Department, Penn State University, 2007, <http://www.cse.psu.edu/~rcollins/CSE486/>.

[2] *Cyganek, B. Siebert, J.* An Introduction to 3D Computer Vision Techniques and Algorithms, John Wiley & Sons, 2009, pp. 194-196, 234, 245.

[3] *Harris, C. Stephens, M.* A Combined Corner and Edge Detector, In: Proc. Alvey Vision Conf., Univ. Manchester, 1988, pp. 147-151.

[4] *Parks, D. Gravel, J.* Corner Detectors, <http://kiwi.cs.dal.ca/~dparks/CornerDetection/index.htm>.

[5] *Nikolova, I. Nikolov, A. Zapryanov, G.* Depth Estimation using Shifted Digital Still Camera, International Conference on Computer Systems and Technologies, 2011 (to be published).

[6] *Noble, A.* Descriptions of Image Surfaces, PhD thesis, Department of Engineering Science, Oxford University 1989, p45.

[7] *Scharstein, D.* View Synthesis Using Stereo Vision, Cornell University, 1997.

[8] *Николов, А.* Определяне на разстояние до обекти в сцена на основата на стереозрение, дипломна работа за придобиване на ОКС магистър, Факултет по Компютърни системи и управление, ТУ-София, 6 юли 2010 г., стр. 35-46, 55-63.