Fakulta Matematiky, Fyziky a Informatiky
Univerzity Komenského
v Bratislave

Diplomová práca

2005 **Michal Jančošek**

Fakulta Matematiky, Fyziky a Informatiky
Univerzity Komenského
v Bratislave

Feature Detection and Tentative Correspondence Estimation in Wide Baseline Stereo

Diplomová práca

Autor : Michal Jančošek

Vedúci diplomovej práce : Doc. RNDr. Andrej Ferko, PhD.

Apríl 2005

Čestne prehlasujem, že predkladanú diplomovú prácu som vypracoval samostatne s použitím prameňov uvedených v zozname na konci práce.

V Bratislave, Apríl 2005 .....

Michal Jančošek

Na tomto mieste sa chcem úprimne poďakovať Doc. RNDr. Andrejovi Ferkovi, PhD. (FMFI – UK, Bratislava) za jeho odborné vedenie, konzultácie, cenné rady a pripomienky, ktoré viedli k vypracovaniu tejto diplomovej práce.

Ďalej by som sa chcel poďakovať Dr. Ing. Jiřímu Matasovi, Ing. Michalovi Perďochovi a Ing. Štěpánovi Obdržálkovi (CMP - The Center for Machine Perception, Praha), za ochotnú komunikáciu a poskytnutie cenných vysvetlení, myšlienok a testovacích dát.

Vďaka patrí aj Joachimovi Bauerovi Dipl.-Ing. (VRVis, Graz) za dodanie testovacích dát.

Table of Content

List of figures

List of abbreviations

CG	Centre of gravity
DR	Distinguished Region
EG	Epipolar geometry
LAF	Local Affine Frame
LoG	Laplacian-of-Gausians
MR	Measurement Region
MSER	Maximally Stable Extremal Regions
NCC	Intensity-Normalized Cross-Correlation
RANSAC	RANdom SAmple Consensus
SIFT	Scale Invariant Feature Transform
TTC	True Tentative Correspondences
WBS	Wide Baseline Stereo

Abstract

The aim of this work is to estimate tentative correspondence in a wide baseline image pairs with a well known method and possibly propose a new method. We made a research on known correspondence methods. This work experimentally shows a verification of our implementation of Maximally Stable Extremal Regions (MSER) detection method and describe our implementation in details. We have also implemented the method to estimate tentative correspondences using Local Affine Frames (LAF). In this work is also proposed a new method called True Tentative Correspondences (TTC) which use MSER as features which were put into correspondence. Input to our algorithm is a widebaseline image pair and output is a set where one element consist of eight tentative correspondences between detected MSER regions, these are the best candidates to compute epipolar geometry between images. There is also proposed a new algorithm to estimate epipolar geometry in image pair using TTC as is mentioned in a future work.

Abstrakt

Táto práca je zameraná na získavanie predbežných korešpondencií v pároch obrázkov scény s veľkou vzdialenosťou medzi polohami fotoaparátov. Je tu navrhnutá nová metóda nazvaná Pravé Predbežné Korešpondencie, ktorá hľadá predbežné korešpondencie medzi Maximálne Stabilnými Extremálnymi regiónmi. Vstup do nášho algoritmu je pár obrázkov scény a výstup je množina, kde každý prvok je osmica predbežných korešpondencií. Tieto osmice sú najlepší kandidáti na výpočet epipolárnej geometrie medzi danými obrázkami. V tomto článku je tiež ukázané experimentálne overenie našej implementácie metódy na detekciu Maximálne Stabilných Extremálnych Regiónov. Okrem toho sme overili metódu na získanie predbežných korešpondencií pomocou Lokálne Afinných Rámcov a navrhli a implementovali jej modifikáciu. V záverečnej časti sme načrtli myšlienku na vytvorenie algoritmu využívajúceho Pravé Predbežné Korešpondencie na získanie epipolárnej geometrie medzi danými obrázkami ako našu budúcu prácu.

1 Introduction

If the luminance I of the point P_A in image A and the point P_B in image B have been defined by the same scene point, we say that P_A and P_B correspond. This is the definition of **Geometric Correspondence** [7].

The correspondence of the feature points in digital image pairs plays an important role in many applications. This type of correspondence is needed to compute relative camera orientation as the first step in process of 3D image synthesis.

Applications, which use the 3D image synthesis, to reconstruct architectonical objects, objects of art, such as building facades, sculptures, fountains etc. are usually used. Data from these applications can be used for example in the projects of virtual cities, like Virtual Heart of Central Europe at www.vhce.info

Epipolar geometry (EG) defines basic geometry between two views. This geometry is use to compute relative camera orientation from some number of correspondence pairs. In this work 8 points algorithm is used. This algorithm is implemented in OpenCV library.

After the step of computation the relative camera orientation, we need to get a point cloud, it means, to find the maximum number of correspondence pairs, lines and regions. This point cloud is used to represent 3D scene. In this process, the known EG is used.

The Stereo problem is the problem of establishing geometric correspondences in a pair of images. The case of the stereo problem of the images taken from two cameras which are close to each other in relation to the viewed scene is called Short Baseline Stereo. There is a large body of literature dealing with this subject (e.g. [8]). The backbone of all of these methods is the Intensity Cross-Correlation [1].

The case of the Stereo problem of two possibly different cameras which are not close to each other in relation to the viewed scene and in a different illumination condition is called Wide Baseline Stereo (WBS).

Majority of methods dealing with WBS problem do need to detect affine invariant features first [2, 3, 4, 6 and 11]. If these features are detected, a matching technique is presented to establish a tentative correspondence.

2 Related work

In this part the overall summary of related work in a field of the WBS is described. The first section 2.1 decribes affine invariation inevitability of detected features and their role in the WBS. In sections 2.1.1 – 2.1.4, according to our opinion, the most common and representative affine invariant feature detection methods are explained. In sections 2.2 and 2.3 the different ways of solving the WBS problem are depicted. Finally in section 2.4 the advantages and disadvantages of the described methods in context of the WBS and the aim of our work are summarized.

2.1 Affine invariant features

The crucial issue in the WBS is to detect visible correspondences between the features. We need to detect the invariant features on the image which are automatically deformed with changing a viewpoint as to keep on covering identical physical parts of scene. The invariance makes them immune against changes in a viewpoint or illumination. So we need to detect features like shapes, corners, regions, lines, which are invariant under affine transformations like rotation, scaling, intensity changes, independently in each of the images. That offers us a powerful tool of correspondence detection between different views of scene.

Perspective projection is not an affine transformation, therefore between two corresponding image patches on the both perspective images there does not have to be an affine relation. There is a question why we are looking for the affine invariant features: In our opinion the following quotation offers us a sufficient answer and is valid in many other works dealing with the WBS problem. Quotation from [16] :

"In this work, an assumption is made, that image deformations can be reasonably well approximated by the local affine transformations of both the geometry and the illumination. Such assumption holds for objects where locally planar surface regions can be found, and where the size of such regions is small relative to the camera distance, so that perspective distortions can be neglected."

The local frame of reference of the detected affine invariant feature (Distinguished Region (DR) [6]) is usually defined by a transformation invariant construction. The DR may be characterised by invariant measurements computed on any part of image specified in the local (DR-centric) frame of reference. In [6] this part of image is called Measurement Region (MR) and algorithms proposed in the literature [3, 4, 6, 10, 11]

use strategies with a similar structure. The core this structure is summarised in following four steps [6]:

1. Detect distinguished regions
2. Describe DRs with invariants computed on measurement regions
3. Establish tentative correspondences of DRs
4. Estimate epipolar geometry in a hypothesise - verify loop

The most common method in establishing a tentative correspondences step (3) selects mutually nearest pairs in the Mahalanobis distance between some descriptors of measurement regions. Afterwards the Cross-Correlation is also usually used to reject low-score matches. The most common method to estimate the epipolar geometry is RANdom SAmple Consensus (RANSAC) [18] which enables the selection of the inliers. This step is explained e.g. in [13]. Therefore the uniqueness of each of these algorithms lies on the affine invariant feature (DR) detection method.

2.1.1 Harris Corner and Edge Detector

According to our opinion the most quoted method in WBS articles and the basic method to detect corner and edge features on image is a Harris Corner and Edge Detector [9]. The method is based on the local auto-correlation function. The main idea of the method is that we look through a small window of an image and if there is a corner or edge then shifting of the window in any direction will be followed by a large change of intensity. As a shifting window the Gausian Smooth Circular Window has been chosen: $w(u,v) = \exp-(u^2+v^2)/2\sigma^2$ Change of intensity of the shift [u,v] is:

$$E(x,y) = \sum_{u,v} w(u,v)\left[I(x+u,y+v) - I(u,v)\right]^2 = \sum_{u,v} w(u,v)\left[xX + yY + O(x^2,y^2)\right]^2,$$

where the first gradient is approximated by:

$$X = I \otimes (-1,0,1) = \partial I/\partial x, Y = I \otimes (-1,0,1)^T = \partial I/\partial y$$

and for small shifts E can be written as:

$$E(x,y) = Ax^2 + 2Cxy + By^2 = (x,y)\begin{bmatrix} A & C \\ C & B \end{bmatrix}(x,y)^T = (x,y)M(x,y)^T,$$

where $A = X^2 \otimes w, B = Y^2 \otimes w, C = (XY) \otimes w$. Then E is closely related to the local autocorrelation function, where M decribes its shape at the origin. This matrix is called Second Moment Matrix or Auto-Correlation Matrix. Let α, β be the eigen values of M which are proportional to the principal curvatures of the local autocorrelation function and also form the rotationally invariant description of M. Measure of corner response is a function of α, β alone, on grounds of rotational invariance:

$$R = Det(M) - kTr(M)^2, Tr(M) = \alpha + \beta = A + B, Det(M) = \alpha\beta = AB - C^2$$

(k - empirical constant) In Figure 1 relations and meanings between α, β and R are shown.

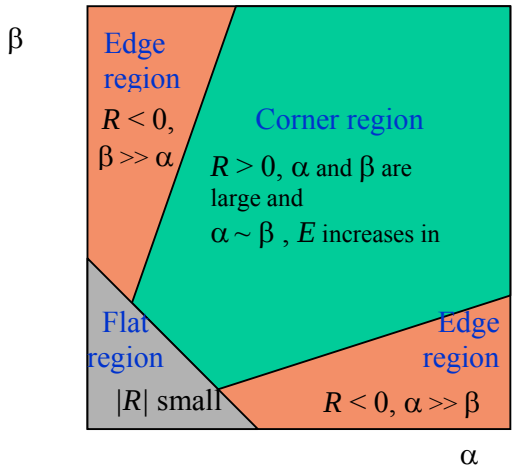

Figure 1: The graph of relations between Edges, Corners, Flat regions and α, β and R

The core of the Harris Corner and Edge Detector algorithm:

a) to find points with large corner response function R (bigger than some threshold)

b) to take the points of local maxima of R

The output of this method are corners with following properties :

a) rotation invariance and invariance to affine intensity change

b) non - invariant to image scale

The essential aim of methods dealing with the WBS problem is to detect affine invariant features, so a big disadvantage of Harris Corner and Edge Detector method is its non-invariant to image scale property.

Harris Corner and Edge Detector is usually used in Intensity Cross-Correlation [1] method of Short Baseline Stereo. This technique is based on the neighborhood comparation of feature points through the Intensity Cross-Correlation. As a neighborhood a small window of (2N+1)x(2N+1) pixels centered around the feature point can be taken. For the feature points (x, y) and (x‘, y‘) the similarity measure is obtained, as follows:

$$C = \sum_{i=-N}^{N} \sum_{j=-N}^{N} (I(x-i, y-j) - \bar{I})(I(x'-i, y'-j) - \bar{I}'),$$

where I and I’ are the intensity values at a certain point and $\bar{I}$ and $\bar{I}'$ are the mean intensity values of the considered neighborhood. Usually N = 3 (7 x 7 pixels window).

In the first step feature points in both images (N in the first one and M in the second one) are found by some detection method, e.g. by Harris Corner and Edge Detector. Then a table of N rows and M colls of values of Intensity Cross-Correlation between subsistent points is computed. The last step is the table evaluation procedure. To i-th row j-th col is chosen wich has the biggest value and the i-j points are declared as corresponding points. Because in Short Baseline Stereo the location of the feature can not be changed widely, only the features with similar coordinates in both images are usually compared. This fact can be used for the reduction of combinatorial complexity of the matching.

2.1.2 Harris – Laplacian

Harris – Laplacian Corner Detector [10] eliminates the disadvantage of Harris Corner Detector. The second moment (auto-correlation) matrix form Harris corner detector has been adapted to scale changes to make it independent of the image resolution. The scale adapted second moment matrix is defined by :

$$\mu(x,\sigma_I,\sigma_D)=\begin{bmatrix}\mu_{11} & \mu_{12}\\ \mu_{21} & \mu_{22}\end{bmatrix}=\sigma_D^2 g(\sigma_1)*\begin{bmatrix} I_x^2(x,\sigma_D) & I_x I_y(x,\sigma_D)\\ I_x I_y(x,\sigma_D) & I_y^2(x,\sigma_D)\end{bmatrix},$$

where σ_I is the integration scale and σ_D is the diferentation scale and I_a is the derivative computed in the a direction. The matrix describes the gradient distribution in a local neighborhood of a point. The local derivatives are computed with Gaussian kernels of the size determined by the local scale σ_D. The derivatives are then averaged in the neighborhood of the point by smoothing with a Gaussian window of size (integration scale).

We study circular regions around a point with increasing radius. Out aim is to find corresponding radiuses independently of two corresponding points so that regions of corresponding sizes will look the same in both images.

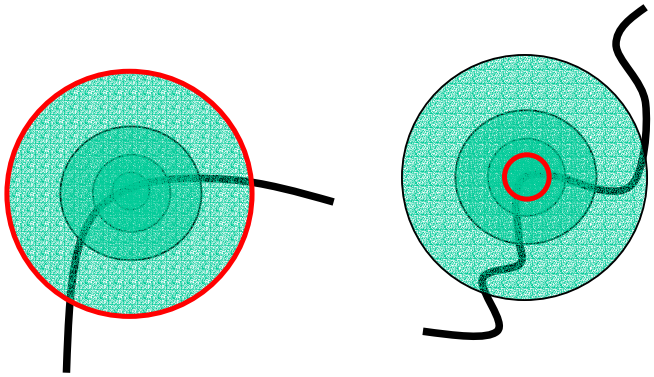

Figure 2: Scale invariant detection [9]

Automatic scale detection is executed by finding point where the normalized Laplacian-of-Gausians (LoG) function is of the maximum value in the point x. LoG. :

$$\left|LoG(x,\sigma_n)\right|=\sigma_n^2\left|I_{xx}(x,\sigma_n)+I_{yy}(x,\sigma_n)\right| .$$

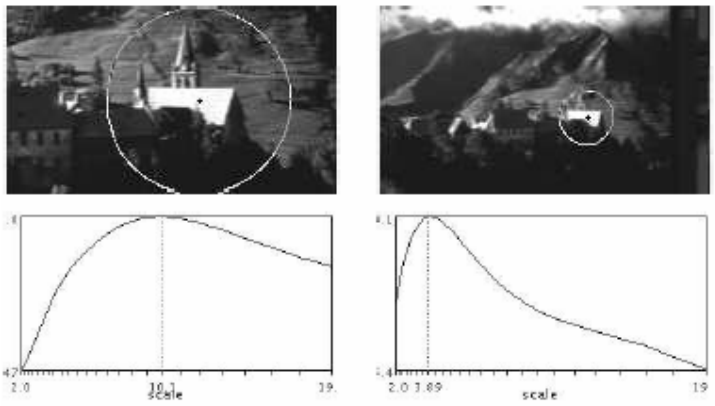

Figure 3: Example of characteristic scales [10]. The top row shows two images taken with different focal lengths. The bottom row shows the response $LoG(x,\sigma_n)$ over scales where. The characteristic scales are 10.1 and 3.89 for the left and right image, respectively. The ratio of scales corresponds to the scale factor (2.5) between the two images. The radius of displayed regions in the top row is equal to 3 times the characteristic scale.

The matrix $\mu(x,\sigma_I,\sigma_D)$ is then computed by integration scale $\sigma_I := \sigma_n$ and the local scale $\sigma_D = s\sigma_n$ where s is a constant factor.

The idea of Scale Invariant Feature Transform (SIFT) [11] is similar to Harris – Laplacian [10] method but for scale detection uses a DoG. Diference-of-Gausian function.

2.1.3 Intensity-Based method

The idea of the Intensity-Based method [3] is based on the analysis of intensity without extraction of features (edges or corners). As anchor point of the method a local extremum in the image intensity is used. Then the intensity function along rays emanating from the extremum is studied: $f(t) = \frac{|I(t) - I_0|}{\max\left(\frac{\int_0^t |I(t) - I_0| dt}{t}, d\right)}$,

where t is the Euclidean arclength along the ray, I(t) the intensity at the position t on the ray, I_0 is the intensity extremum and d is a small number to prevent a division by zero.

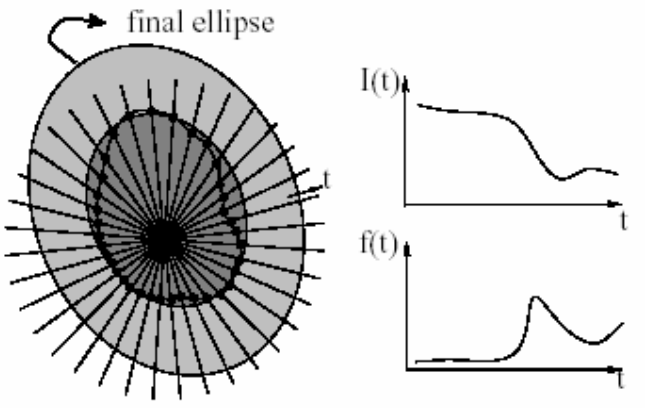

Figure 4: The intensity along "rays" emanating from a local extrema. The point on each ray for which a function f(t) reaches an extremum is selected and these points are linking together to get affinely invariant region, to which an ellipse is fitted using moments.

The position of extrema of f(t) is invariant to the geometric and photometric transformations, next of these points from the same local extremum are linked together to enclose an affinely invariant region (see figure 4). This region is then replaced by an ellipse having the same shape moments up to the second order. This ellipse is affinelly invariant but its centre is not the same as the original anchor point (the intensity extremum).

2.1.4 Maximally stable extremal regions

Next method to detect affine invariant features is a method of Maximally Stable Extremal Regions [6] . The idea of this method is informally explained in [6] as follows. Let us assume all possible thresholdings of a gray-level image I through S={0,1,…,255}. We will mark all pixels below threshold i as white coloured ones and the rest as black coloured ones. The result made of 256 tresholded images, will be a movie. The first image of the movie will be white and then black regions belonging to local intensity minima will appear and these will grow consequently. At some point regions correponding to two local intensity minima will merge. The last image at the end of the movie, will be black. The set of all connected components of all frames of the movie is the set of all extremal regions. Afterwards we will select these extremal regions which support stays virtually unchanged over a range of threshold. Selected regions were designated as Maxilmally Stable Extremal Regions. MSER has a following properties :

- Invariance of affinne transformation of image intensities
- Covariance to adjacency preserving (continuous) transformation T: D->D on the image domain
- Stability, since only extremal regions whose support is virtually unchanged over a range of thresholds is selected
- Multi-scale detection. Since no smoothing is involved, both very fine and very large structure are detected
- The set of all extremal regions can be enumerated in O(n log log n), where n is the number of the pixels

2.2 Vanishing points

The method [2] exploiting the Vanishing points is based on the fact that two parallel lines in 3D space can intersect after their perspective projection to the projective plane. This intersection is called Vanishnig Point. In the perspective projection of the cube there can be three vanishing points. This method has been designed for reconstruction of architectural objects. It has several prerequisites and assumes that the building was built along three orthogonal axes.

The first step is to extract straight lines from the two images. In the second step, the vanishing points are detected for each image separately. Consequently the lines, overcapitalization of which isn’t in neighborhood of one of the detected vanishing

points, are excluded. In the third step, edges are intersected to points in image space, and point correspondence and relative orientation are detected simultaneously.

2.3 Triangulation based correspondence

Kolingerova et al. presented in [19] a different idea of feature matching based on comparision of two triangulations of feature poins detected on left and right image.

2.4 Conclusions

In section 3.1 the meaning of the affine invariant feature is descibed. In 3.1.1-3.1.2 the methods which detect first kind of feature - the corners are shown. In 3.1.1 Harris Corner and Edge Detector is decribed. This method detects corner features which are invariant only under rotation and intensity change. The use of Harris Corner and Edge Detector dealing with Near Baseline Stereo problem is also shown there. Harris Corener and Edge Detector is the base of the other two methods Harris - Lapalcian and SIFT described in 3.1.2. These have added the scale invariant property to detected corner features.

Another kind of features are regions detected by intensity based method described in 3.1.3. This method produces features which are invariant under all affine transformations. Another region based method is described in 3.1.4.

All of the features described in sections 3.1.2 – 3.1.4 are used as input to methods dealing with WBS problem and can be used to reconstruct scenes from different environments. These are, according to our oppinion, the best nowadays.

Then in 3.2 we show another way of solving the WBS problem which is based on vanishing points detection and, according to our opinion, there is a big disadvantage because of its prerequisite that buildings were built along three orthogonal axes and this method can be applied only in architectural image pairs. In general we assume that architectural objects do not have to be built along three orthogonal axes as sculptures, fountains and also many modern buildings. Therefore we think that Vanishing Points Method is for our goal significantly useless. Finally, in section 3.3 we describe the idea of the use of triangulation in correpondence estimation process.

We have decided to verify MSER method [4]. The reasons why we have decided for this method are:

a) We are interested in MSER method idea because of its simplicity

b) In [15] the experiments established the superior performance of the MSER detector

c) Also in [12] the MSER detector was one of the best local detector

3 MSER detection method [4] verification

In the section 3.1 there are some basic definitions, then in the section 3.2 our MSER detection process in details, including our invention in MSER detection function, is described. Finally in 3.3 there are the results of experimental verification of our MSER detection implementation and a comparison of experiments results of our modification of MSER detection function.

3.1 Basic definitions

These are the definitions from [4] on which our implementation of MSER method is based:

Def 1. Image I is mapping $I : D \subset Z^2 \to S$. Extremal regions are well defined on images if:

1) S is totally ordered, i.e. reflexive, antisymetric and transitive binary relation ≤ exists. In our work only S={0,1,2, …, 255} is considered.
2) An adjacency (neighbourhood) relation $A \subset D \times D$ is defined. In this work 4-neighbourhoods are used, i.e. $p, q \in D$ are adjacted $(pAq) iff \sum_{i=1}^{d} |p_i - q_i| \leq 1$.

Def 2. Region Q is a contiguous subset of D, i.e. for each $p, q \in Q$ where is a sequence $p, a_1, a_2, ..., a_n, q$ and $pAa_1, a_i A a_{i+1}, a_n Aq$

Def 3. (Outer) region boundary $\partial Q = \{q \in D \setminus Q : \exists p \in Q : qAp\}$ i.e. the boundary of Q is the set of pixels being adjacted to at least one pixel of Q but not belonging to Q.

Def 4. Extremal region $Q \subset D$ is a region such that for all $p \in Q, p \in \partial Q : I(p) > I(q)$ (maximum intensity region) or $I(p) < I(q)$ (minimum intensity region).

Def 5. Maximally Stable Extremal Region.

Let $Q_1, ..., Q_{i-1}, Q_i, ...$ be a sequence of nested extremal regions, i.e. $Q_i \subset Q_{i+1}$.

Extremal region Q_i is maximally stable if $q(i) = |Q_{i+\Delta} \setminus Q_{i-\Delta}| / |Q_i|$ has a local minimum at i* (|.| denotes cardinality). $\Delta \in S$ is a parameter of the method.

3.2 Our MSER detection process overview

Maximally Stable Extremal Regions are decribed in the section 2.1.4. In the section 3.1 the basic definitions of MSER are presented. Then MSER detection process is described in details. In the section 3.2.1 the regions tree structure and its detection is explained, then in the section 3.2.2 its traversing is explained and with help of function defined in 3.2.3 and in the section 3.2.4 our MSER detection function is defined. Finally MSER detection process is explained in section 3.2.5.

3.2.1 Region tree detection

The first step is to sort pixels by intensity. Then Algorithm 1 goes through the image intensities from 0 to 255 where the forest structure with following properties is build simultaneously.

a) Each node represents the pixels with the same gray value only

b) Each node n can have a parent node representing the pixels with the same or bigger gray value than the gray value of the pixels of the node n

c) If node n has a parent node representing the pixels with the same gray value as the gray value of the pixels of the node n, then this is marked in the rename array

As shown in Algorithm 1 there are four cases of processing of the pixel pix at some gray level (we assume 4-neighbourhood). By every neighbour of pix we get its node (if the pixel is assigned to some node) and we find the highest parent of its node.

a) when the pix does not have any neighbour pixels which belong to some node, then a new leaf node is created.

b) when the pix has some neighbour pixels which belong to the only and highest node n and the pix has the same gray value as the pixels of this node, then the pix is assigned to the node n.

c) when the pix has some neighbour pixels which belong to the only and highest node n and the pix does not have the same gray value as pixels of this node, then the new node is created as a parent node of node n .

d) when the pix has some neighbour pixels which belong to more than one different highest nodes then the new node is created as a parrent node of these neighbour nodes. This merge procedure is shown in Figure 5.

After this step there is one tree as a consequence of merge procedure. This tree is reduced with help of its rename array to get the tree where each node can have only a parrent with pixels of bigger gray value. Then the maximum height of the tree is 256. Now the tree has a following properties :

a) when we get all pixels of a node n and all pixels of its subtree then these pixels represent an extremal region that was deffined in Def 4.

b) when we get the way from any leaf which represent pixels from a local extrema to any node, this way represents a growing process of its region.

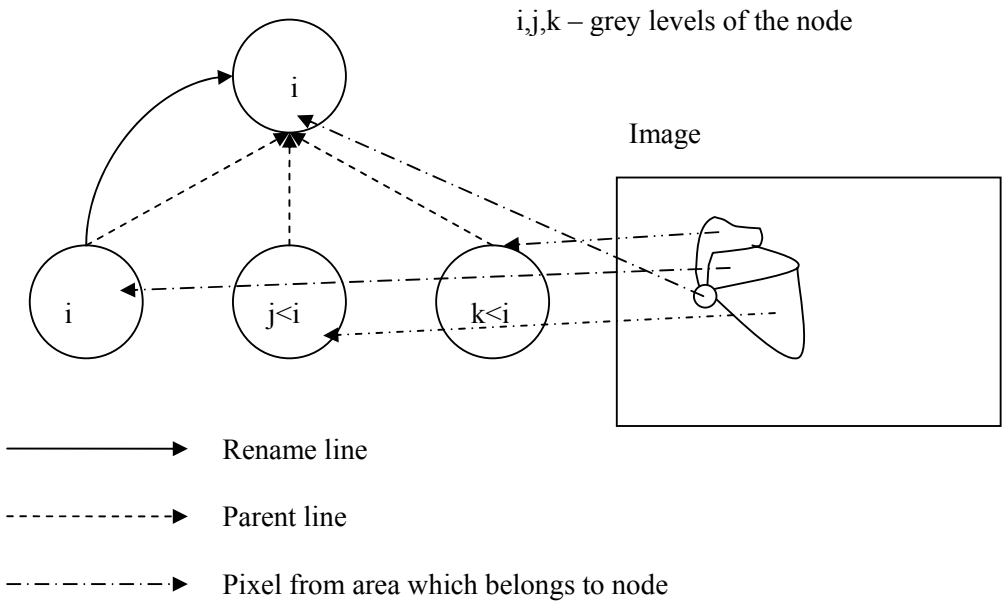

Figure 5: Merging nodes

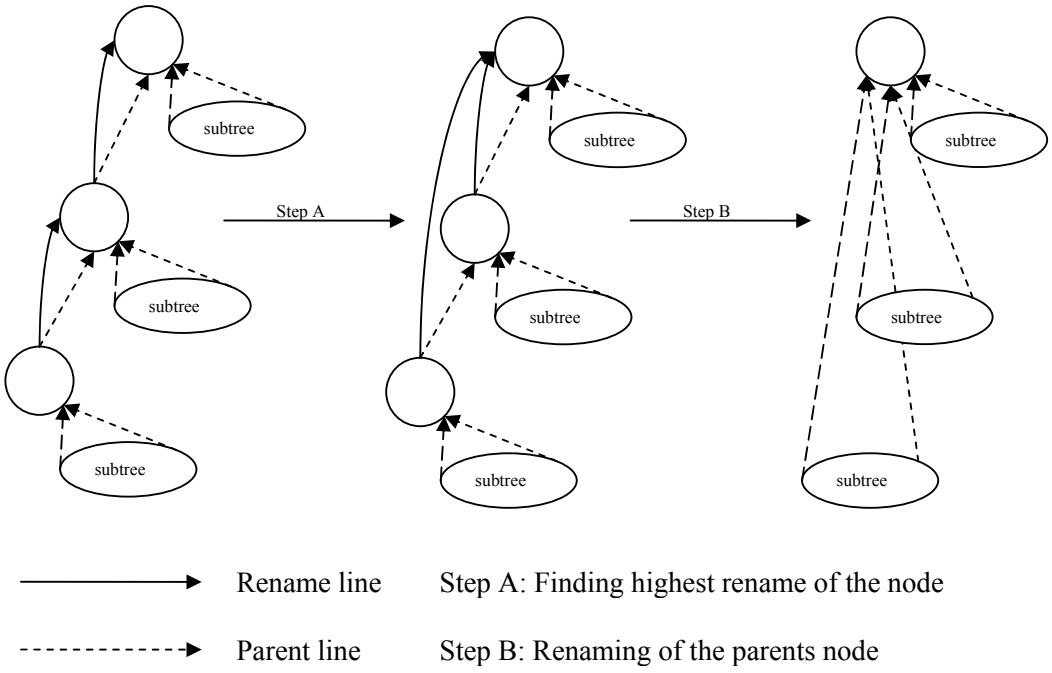

Figure 6: Tree reduction

Algorithm 1: regions tree building

```
Input
Output
01: SORT pixel in increasing order of gray values
02: for graylevel = 0 to 255 do
03:    for all pixels pix from GrayValuePixels[GrayLevel] do
04:                  for all neighbours q of pix do
05:                  find highest parent of q
06:                  put him into topNeighs field if isn't  there
07:           end for
08:           (* creation of new node - leaf*)
09:           if neigbCount = 0 then
10:                  parents->add(-1); rename->add(-1);
12:                  graylevels->add(graylevel)
13:                  areas->add(1)
14:                  leaves->add(parents->high)
15:                  image[pix]->nodeIndex = parents->high
16:           end if
17:           (* adding pixel to existing node *)
18:           if neigbCount = 1 and pix->GrayLevel = topNeigs[0] then
19:                  areas[topNeigs[0]] += 1
20:                  image[pix]->nodeIndex = topNeigs[0]
21:           end if
22:           (* adding pixel to a new parrent node with one child*)
23:           if neigbCount = 1 and pix->GrayLevel > topNeigs[0] then
24:                  parents->add(-1); rename->add(-1);
25:                  graylevels->add(graylevel)
26:                  areas->add(1)
27:                  image[pix]->nodeIndex = parents->high
28:                  parents[topNeigs[0]] = parents->high
29:           end if
30:           (* merging more childs to new node *)
31:           if neigbCount > 1 then
32:                  parents->add(-1); rename->add(-1);
33:                  graylevels->add(graylevel)
34:                  areas->add(1)
35:                  image[pix]->nodeIndex = parents->high
36:                  for all neigh from topNeighs do
37:                        areas[parents->high] += areas[neigh]
38:                        parents[neigh] = parents->high
39:                        if graylevels[neigh] = graylevel then
40:                              rename[neigh] = parents->high
41:                        end if
42:                  end for
43:           end if
44:    end for
45: end for
46: (* tree reduction *)
47: (* finding highest rename node *)
48: for nodeIndex = 0 to parents->length do
50:    newNodeIndex = nodeIndex;
51:    while rename[newNodeIndex] != -1  do
52:           newNodeIndex = rename[newNodeIndex]
53:    end while
54:    rename[nodeIndex] = newNodeIndex
57: end for
58: (* renaming of the parrents = tree reduction *)
59: for nodeIndex = 0 to parrents->length do
60:    if parrents[nodeIndex] != -1 and
rename[parents[nodeIndex]] != -1 then
61:           parents[nodeIndex] = rename[parents[nodeIndex]]
62:    end if
63: end for
64: (* renaming of the image pixels *)
65: for all pixels pix from image do
66:    image[pix]->nodeIndex = rename[pix->nodeIndex]
67: end for
```

3.2.2 Regions tree traversing

Now we should make an MSER detection step. The bottom-up algorithm goes through the field of leaves. For each leaf l, we have to find the way w, where every node n has following maximal property :

If we get the node n, which is the one of the parent's p children, then the node n has to represent an extremal region with maximal area of all region areas represented by the rest of p children.

It is equal to the situation when we assume our tree, where the maximal node is on the left. We have to traverse our tree by preorder up-bottom Algorithm 2. The bottom-up approach is simplier because of the fact that we do not have to remember children of each node.

Algorithm 2: Tree traversing (up-bottom approach)

```
01: FIFO q
02: TraveseTree(root)
03:
04: procedure TraverseTree(node)
05:    q->Add(node)
06:    if not IsLeaf(node)
07:           for all children of node do
08:                  TraverseTree(child)
09:           end for
10:    else if
11:           FindMSERRegions(q)
12:           q->Clear()
13:    end if
14: end procedure
```

The maximal property of the way w is in [4] described as "A merge of two components is viewed as termination of existence of the smaller component and an insertion of all pixels of the smaller component into the larger one.". Thus we have to detect MSER regions on the small component and later on the larger one.

3.2.3 MSER region detection function

When we have a way w which represents the growth of the region, for each node n we have to compute some value by the Definition 5. In this definition some delta constant is used. We wrote to the author to get its value. The result of our comunication is that author's team does not use the function as it is publicated in [4], we were told that they are using this function now :

Def 6. Let $Q_1,...,Q_{i-1},Q_i,....$ be a sequence of nested extremal regions, i.e. $Q_i \subset Q_{i+1}$. Extremal region Q_i is maximally stable if

$q(i) = j - i; j = \max\{k \mid |Q_k| - |Q_i| < 2 * sqrt(|Q_i|)\}$ has a local maximum at i* (|.| denotes cardinality) and $q(i) > \Delta; \Delta \in S$

is a parameter of the method (in our experiments $\Delta := 10$).

This MSER detection function for Region Q_i deals with: how long the region must grow till its area grows more than his circumference, where $2 * sqrt(|Q_i|)$ is some approximation of real circumference of the region Q_i .

This detection function gives better MSER regions in experiments for tentative correspondence estimation.

3.2.4 MSER regions detection

Detection of MSER regions is done by FindMSERRegions procedure, where the input is a way w which represents the growth of the region.

The length of the way is maximally 256. The sequence of nodes on the way is : $\{n_0, n_2, \ldots, n_h\}$, where h = Length(w)-1 and represent the sequence of extremal regions defined in Definition 6 (7).

Then for each node n_i we compute some value v_i by function which is defined in Def 6 (7). Now there is the array of values $\{v_0, v_2, \ldots, v_h\}$ according to the Definition 6 (7) of MSER region we have to find some local maximums in this array to get MSER regions.

In comunication with Dr. Matas' team (Michal Perdoch) we got the information that these values are also filtered. This filter goes through the array of local maximum of values and if areas in i-th and i+1 –th local maximum are in less then 10% difference they are linked together.

The regions belonging to remaining local maximums are then marked as MSER regions.

3.3 MSER detection experiments

The Dr. Matas' team has given us the MSER binary file in which MSER detection using the function from Definition 6 is implemented. The input to this binary file is the picture and some paramaters and output is built up by detected MSER regions. This means a powerful tool for our reimplementation of MSER detection method results verification. The comparison is done by comparing each region detected by our implementation with each region detected by author's binary file.

The experiments are realised on two imagesets.

1. Images from Michal Perdoch, CMP, Prague
2. Images from Joachim Bauer, VRVis, Graz

Image set	# orginal MSER	# our MSER	# identhical	% match
Prague	9462	9403	9003	95%
Graz	4667	4610	4352	93%
	14129	14013	13355	95%

Table 1: Verification of correctness of our implementation of MSER detection method.

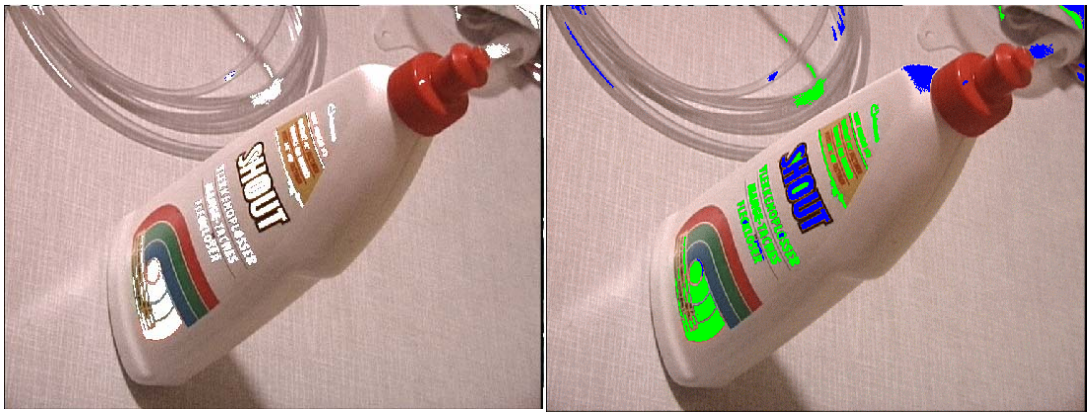

Figure 7: Left: Identical regions, Right: Our MSER regions (blue: MSER-, green: MSER+)

3.4 Conclusions

As you can see in the Table 1 the experiments show: original and our MSER detection implementations take on 95% identical results. According to our opinion the reason of 5% mismathes are in constant values and comparison > vs. >= in proposed algorithms. We can claim that our reimplemantation of MSER detection method has been succsessfull.

4 Tentative correspondence estimation process

The next step after MSER region detection is to find Local Affine Frames (LAF) of three types. In the section 4.1 there are the definitions and the explanation of these three types of LAF and its normalisation. In the section 4.3 there are some implemetation details, then in 5.4 tentative correspondence estimation process using LAF is described. In 5.5 experiments are provided and in the section 5.6 there are conclusions.

4.1 Local frames of reference and normalization

Local affine frame can be viewed as local coordinate system which is detected invariantly to both affine transformations (geometric and illumination). Local affine frames are used to provide normalisation of image patches into canonical frame to enable direct comparision with Intensity Normalised Cross-Correlation Method. It might not be possible to construct LAF for every MSER. For example if there is an elliptical MSER it is viewed as affine transformation of circle and there is no dominant direction because the circle is completely isotropic. On the other hand for some MSER multiple LAFs in a stable and thus repeatable way can be affine-invariantly constructed.

Def 8. Centre of gravity (CG) of a region is $\mu = \frac{1}{|Q|}\int_Q x dQ$.

Def 9. Covariance matrix of region Q is $n \times n$ matrix defined as

$$\sum = \frac{1}{|Q|}\int_Q (x-\mu)(x-\mu)^T dQ$$

Def 10. Bi-tangent is a line segment bringing a concavity, i.e. its endpoints are both on the regions outer boundary and the convex hull, all other points are part of the convex hull.

Local Affine Frame is the set of three points which define the local coordinate system. These three points need to be affine invariant. The first type of LAF is obtained from covariance matrix. From this matrix we obtain properties of ellipse E which aproximates the detected MSER.

a) Centre of gravity of region is the centre of ellipse E
b) Eigen vectors of covariance matrix are the directions of both ellipse axises
c) Eigen values v_1 and v_2 of this matrix define the length of both ellipse axises as : $a = 2 * sqrt(v_1), b = 2 * sqrt(v_2)$

Affine covariance of CG and covariance matrix is shown in [17]. Transformation by the square root of inverse of the covariance matrix normalises the ellipse E to unit circle and defines the transformation from local coordinate system defined with ellipse parameters to new global, but this normalises the MSER region up to a known rotation. Thus we have to complete the affine frame to resolve the rotation ambiguity. In our work the following directions have been used:

1. centre of gravity to a contour point of globally maximal distance from the CG (LAFType1)
2. centre of gravity to a contour point of globally minimal distance from the CG (LAFType2)
3. centre of gravity to a contour point of locally maximal distance from the CG (LAFType3)
4. centre of gravity to a contour point of locally minimal distance from the CG (LAFType4)

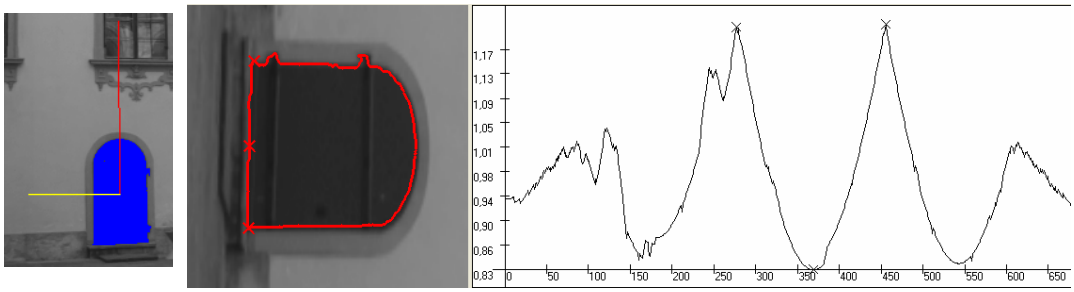

Figure 8: On the left : Ellipse axeses defined by covariance matrix with the scale factor 3, On the right : Normalized region with detected extremal points and its curvature.

The second type of LAF is based on bitangent, the two tangent points are combined with the third point to complete an affine frame.

1. the most distant point of the concavity part from the bitangent (LAFType5)
2. the most distant point of MSER from the bitangent (LAFType6)
3. the CG of MSER (LAFType7)

The invariance of bitangents is the consequence of the affine invariance of the convex hull construction. The invarinace of the third points was shown in [16].

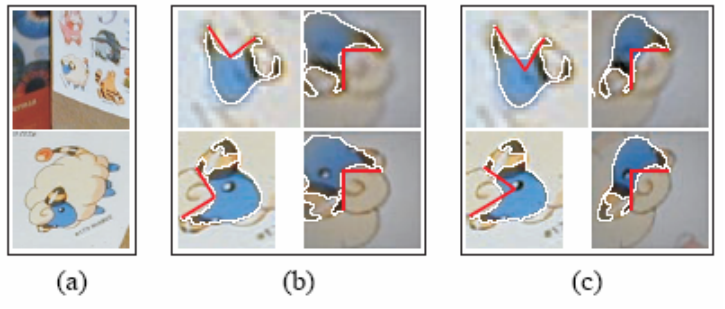

Figure 9: a) Two original images, b) Left : bitangent - max. distant concavity point Right : normalized, c) Left : bitangent - centre of gravity point Right : normalized

When there is the LAF we could compute an affine transformation mapping the LAF to a normalised coordinate system, then transform the image part containing MSER, to which the LAF belongs, to normalised coordinate system.

4.2 Implementation details on LAF detection

The problem of implementation of the first type of the LAF was to detect the local extremums of curvature.

We contacted the author Obdrzalek to get more information about type of curvature and local extremum detection proposed in articles [16, 17]. We were told to us the curvature consisting of distances from the centrum of gravity of the region, and to use the Non Maxima Suppresion algorithm to choose the local extrema.

It was also confirmed by the author how do they detect the bitangents, because in the article only detection of bitangents, which lie on the convex hull, is described but the region can have more bitangents which do not lie on the convex hull of the region. This fact has been used in the implemantation of the last two types of LAF.

4.3 Tentative correspondence estimation of MSER using LAF

Tentative coreespondence estimation si based on Intensity-Normalized Cross-Correlation (NCC) [5] between two normalized region in terms of the local coordinate frames. $NCC(I_1, I_2) = \frac{\sum_x (I_1(x) - \overline{I_1})(I_2(x) - \overline{I_2})}{\sqrt{\sum_x (I_1(x) - \overline{I_1})^2 \sum_x (I_2(x) - \overline{I_2})^2}}$,

where $\overline{I_1} = \frac{1}{N}\sum_x I_1(x), \overline{I_2} = \frac{1}{N}\sum_x I_2(x)$ are the means of windows I_1 and I_2. NCC takes on values in [-1,1] (1 being the most similar, -1 being the least similar) and is invariant to illumination transformations such as contrast and brightness modifications. We keep the value –(NCC-1) then 0 being the most simillar and 2 being the least simillar.

In the case we want to compare two corresponding frames of the left and of the right image in normalised coordinate system there is a need that these frames represent a parts of some plane in real world, if they do not, the images should be different with the change of the viewpoint. This is **the local planarity assumption**. In general, the MSER has a pure histogram and if we put into NCC two MSER without their surroundings the result will be delusive. So we need to get MSER with some surroundings. Because the LAF is represented by three points of MSER we want to preserve surroundings of all of them. We choose the sqaure 1*s x 1*s in normalised coordinate system with the centre in point [0.5, 0.5] where s is some scale factor. We can name it normalised measurement region as in [16]. Then we will do the resampling of the intensities (using billinear interpolation) of the LAF's MR into a raster into the normalised coordinate system. To represent the content of normalised MR we use rasters of size NWxNW pixels.

In the section 4.3.1 our invention in iterative NCC is explained, in the section 4.3.2 tentative correspondences estimation process is described, in the section 4.3.3 there are results of experiments and finally in 4.3.4 there are some conclusions about tentative correspondence estimation.

4.3.1 Iterative NCC

A larger frames are of higher discriminative potential, but they are more likely to cover an object area that violates the local planarity assumption. So we have decided to get the best result of iterative NCC algorithm on raster from normalised MR. Iterative NCC starts on some scale factor s_1 which belongs to raster :

(iteratNCCBeginVal)x(iteratNCCBeginVal),

(where $iteratNCCBeginVal = (NW / s) * s_1$) and ends with the scale factor s which belongs to raster NW x NW. At first $\overline{I_1} = \frac{1}{N}\sum_x I_1(x), \overline{I_2} = \frac{1}{N}\sum_x I_2(x)$ are computed and $Sum_1 = \sum_x (I_1(x) - \overline{I_1})^2, Sum_2 = \sum_x (I_2(x) - \overline{I_2})^2$ and

$Sum_{12} = \sum_x (I_1(x) - \overline{I_1})(I_2(x) - \overline{I_2})$ for the initial raster. Then $\overline{I_1}, \overline{I_2}, Sum_1, Sum_2, Sum_{12}$ are iterativelly computed for every following i-th frame

(iteratNCCBeginVal+i)x(iteratNCCBeginVal+i) where

$i \in \{0,1,\cdots, NW - iteratNCCBeginVal\}$ In each step is computed

$NCC_i = \frac{Sum_{12}}{\sqrt{Sum_1 * Sum_2}}$, so that there is the array of NCCs. The computation cost of this array is the same as the computation cost of NCC for final raster NWxNW.

Figure 10: On the left there is need of bigger scale factor and on the right there is need of smaller scale factor for two corresponding frames (NCC goes from 0 to 2, SCALE goes from 1,25 to 3)

4.3.2 Comparative process of tentative correspondence estimation

For each two frames of the same type from the left and the right image (iterative) NCC values are calculated between them and saved in the correspondence map.

Some frame pair A,B forms a tentative correspondence if A matches B with the highest correlation from all right frames and B matches A with the highest correlation from all left frames and this correlation is smaller than some threshold (maxCorr). For each frame pair we have 7 (iterative) $NCC_{LAF(i)}$ values for each LAF type we have one (iterative) NCC value, so now there is a need to define :

Def 10. Frame A matches frame B with correlation c between them according to LAF types $t_1,\ldots,t_k$ when $c = \min\{NCC_{LAF(t1)},\cdots,NCC_{LAF(tk)}\}$, where k<=LAFCount.

To get more tentative correspondences in our experiments we compute tentative correspondences in 4 steps for :

1. Each LAF type separattely
2. LAF types 1,2,3,4
3. LAF types 5,6,7
4. All LAF types 1,2,3,4,5,6,7

When we want to add tentative correspondence A,B with correlation c to field of tentative correspondences in each of the steps 2,3,4, to reject multi tentative corespondences for frame A or B we must do the following:

1. If A exists in the field and matches C with correlation corr we compare c and corr and if c < corr then C:=B and corr:=c
2. If B exists in the field and matches C with correlation corr we compare c and corr and if c < corr then C:=A and corr:=c
3. If A exists in the field and matches B with correlation corr we compare c and corr and if c < corr then corr:=c

4.4 Experiments

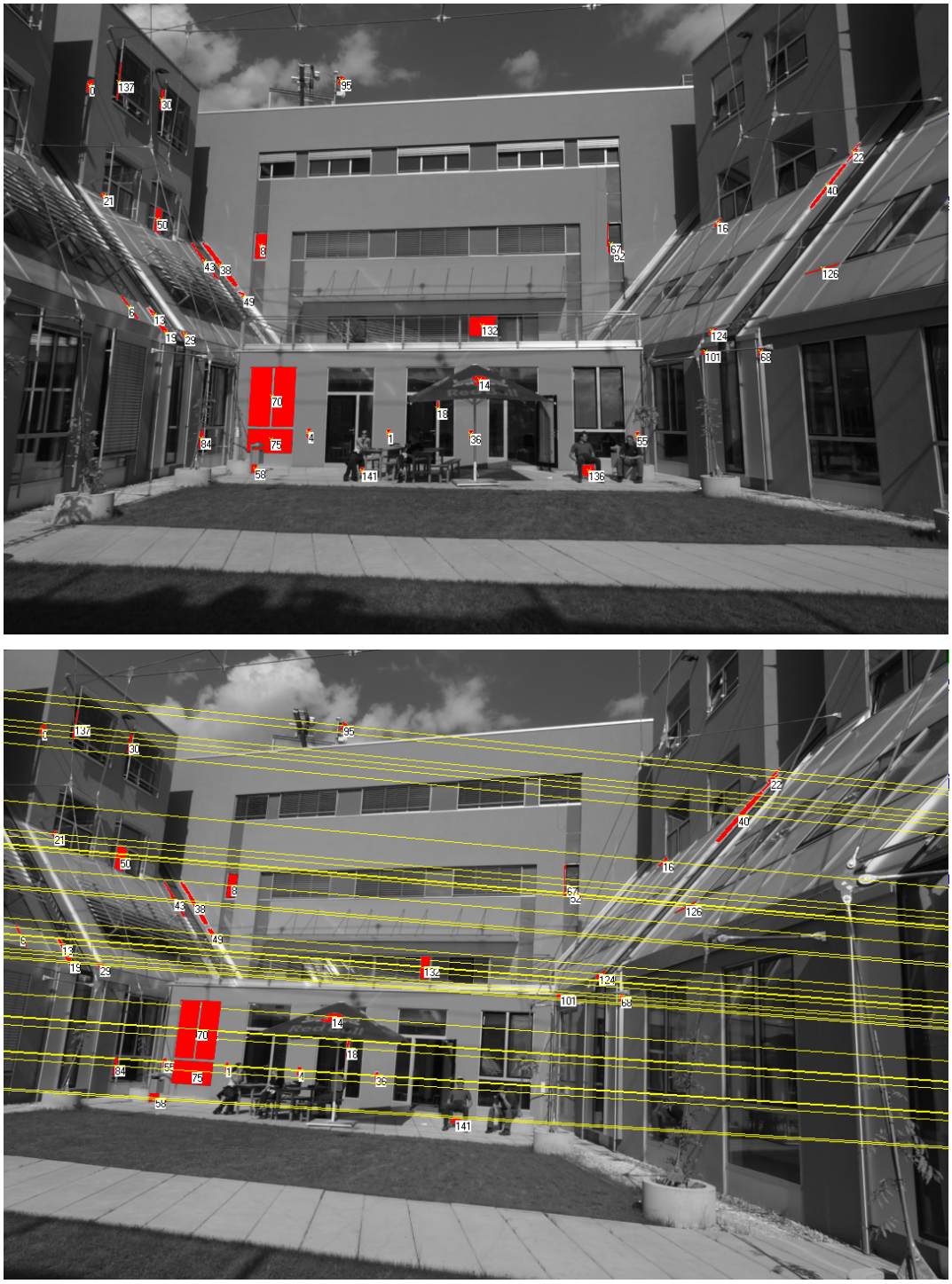

Figure 11: Detected inliers on Mensa02.png (top) and Mensa03.png (bottom) architectural images

Figure 12: Detected inliers on LeafsA.jpg (top) and LeafsB.jpg (bottom) images of nature

		NCC			Iterative NCC		
Left Image	**Right Image**	**# inliers**	**# TC**	**%**	**# inliers**	**# TC**	**%**
bookshA.tif	bookshB.tif	27	77	35%	28	80	35%
castleA.tif	castleB.tif	26	168	15%	30	207	14%
graffA.ppm	graffB.ppm	55	122	45%	55	148	37%
kampaA.tif	kampaB.tif	15	68	22%	14	79	18%
leafsA.jpg	leafsB.jpg	34	172	20%	40	208	19%
plantA.tif	plantB.tif	11	90	12%	9	87	10%
shout1.tif	shout3.tif	47	93	51%	40	95	42%
vbnA.tif	vbnB.tif	10	35	29%	11	35	31%
wallA.jpg	wallB.tif	17	102	17%	15	110	14%
washA.tif	washB.tif	46	99	46%	49	104	47%
chem_lab_01.png	chem_lab_02.png	75	137	55%	74	137	54%
landhaus1.png	landhaus2.png	21	73	29%	19	75	25%
landhaus2.png	landhaus3.png	50	108	46%	60	136	44%
landhaus3.png	landhaus4.png	59	99	60%	63	114	55%
mensa01.png	mensa02.png	77	153	50%	85	180	47%
mensa02.png	mensa03.png	40	143	28%	38	153	25%
temmel01.png	temmel02.png	128	262	49%	117	286	41%
temmel02.png	temmel03.png	132	258	51%	134	286	47%
sum		**870**	**2259**	**39%**	**881**	**2520**	**35%**

Table 2: Experiments on Tentative Correspondence estimation using NCC and Iterative NCC methods.

4.5 Conclusions

In the section 4 there are described two methods to estimate Tentative Correspondences between WBS image pair. The first method is implementation of ideas of the method which is described in articles [16, 17, 20]. As you can see in the Figure 11 and 12 this implementation produces satisfying results. The second one is a modification of the first one but it uses Iterative NCC algorithm instead of normal NCC. As you can see in the Table 2 the use of Iterative NCC method produces only 11 inliers more than clasic NCC method, and the percentage of inliers are approximatelly the same, so we have to state that the contribution of our Iterative NCC method is not so satisfying. Our expectations of better results by the use of the Iterative NCC algorithm was based on theoretical explanation as is mentioned in the section 4.3.1. Our future work is to improve the method.

5 True Tentative Correspondences

In this section a new method called True Tentative Correspondences (TTC) to estimate tentative correspondence in a wide baseline image pairs is described. This method uses MSER as features which were put into correspondence and is based on the LAF and The Sideness Constraint. The input to our algorithm is a widebaseline image pair and the output is a set where one element consists of eight tentative correspondences between detected MSER regions, which are the best candidates to compute epipolar geometry between images.

In the section 5.1 The Sideness Constraint is described. In the next section 5.2 the idea of our new method called True Tentative Correspondence is described and in the section 5.3 the process of estimation True Tentative Correspondences tree. Then in the section 5.4 there are experiments and finally in the section 5.5 there are conclusions.

5.1 The Sideness Constraint

This logical rule (The Sideness Constraint) is described in [14]. Let us consider two corresponding point pairs: L1, L2 on the left image and R1, R2 on the right image. We can divide the left image in to the left and the right part according to directed line $\xrightarrow[L1L2]{}$ and we can do the same operation with the right image according to directed line $\xrightarrow[R1R2]{}$. If there is correspondence pair of points A,B, they have to lie on the same part of the left and the right image.

The function $side(A, L1, L2) = sign((L1 \times L2)A)$ returns 1 if A is on the left side of the directed line $\xrightarrow[L1L2]{}$ and -1 if it is on the right one.

The equation $side(A, L1, L2) = side(B, R1, R2)$ states that A should be on the same side of the line on both of the views.

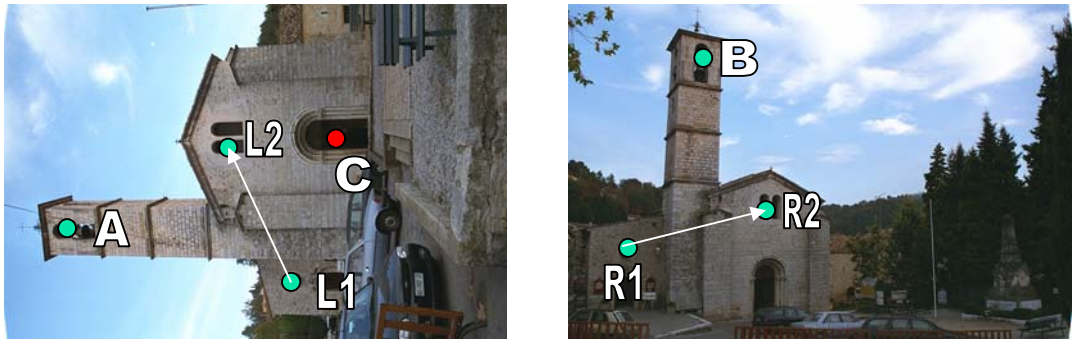

Figure 13: The Sideness Constraint

5.2 True Tentative Correspondences

The essential idea of our method is to search the following correspondence using the best correspondences found before, of which we assume to be geometric correspondences.

Step 1: We take the correpondence regions A_0,B_0 with the best correpondence value computed in comparative process on LAF types 1 and 2. In every experiment we have realised that the best correspondence is the geometric one. Then for each region on the left image we compute LAF with direction to A_0 and for each region on the right image with direction to B_0.

Step 2: The comparative process is running again, then we get the first best different correpondence regions A_1,B_1 with the best correspondence value different from A_0,B_0 which are not closer to A_0,B_0 than 10 pixels on both images separatelly.

Step 3: There are two tentative correspondences and we assume that they are geometric and we can now use The Sideness Constraint. Then for each region on the left image we compute two LAFs with directions to A_0,A_1 and for each region on the right image with directions to B_0,B_1. Afterwards we compute correspondence map and save it to the memory.

Comment: We reject correspondences by The Sideness Constraint by setting correspondence value to a certain big number (5.0) in correspondence map of regions which are on the different sides of the directed line on the left and on the right image.

Step 4: This step is repeated untill n=7:

1. Load correspondence map
2. Reject correspondences by The Sideness Constraint for each pair of correspondences from the True Correspondences Set { A_0,B_0; A_1,B_1; … ; A_n,B_n } by changing the correspondence map
3. Run comparative process on this correspondence map and add the first best different correspondence to the True Correspondences Set

There are 8 correspondences as output from this method which we named True Tentative Correspondences (TTC).

Used rotation direction from CG_1 of $MSER_1$ to CG_2 is invariant under the viewpoint change only if CG_2 lies on the same plane as the $MSER_1$ region to which the rotation ambiguity is resolved (in respect to the local planarity assumption). The mistake depends on the angle between the $MSER_1$ plane and the CG_1 to CG_2 directed line and on the change of camera viewpoint. The smaller is the angle the smaller is the mistake. According to our opinion this mistake is in many cases smaller than mistakes in rotation from curvature analysis because the boundary of MSER region is discrete and under the scale change not so stable.

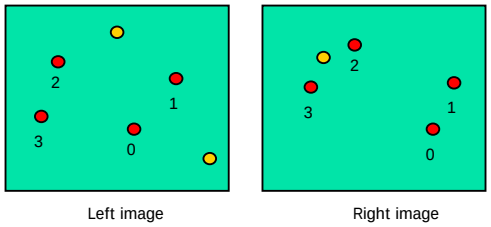

Figure 14: Corresponding regions

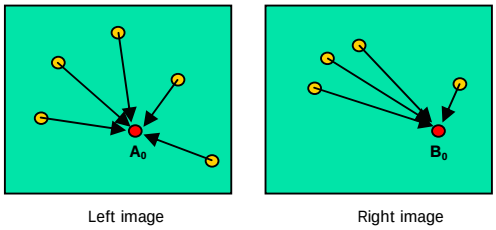

Figure 15: TTC – Step1

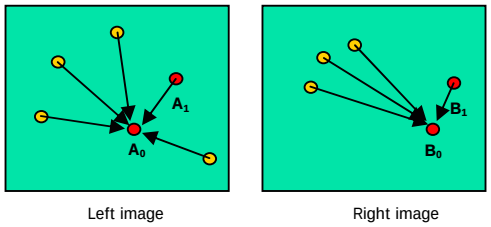

Figure 16: TTC – Step 2

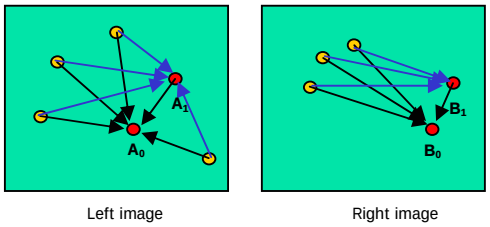

Figure 17: TTC – Step 3

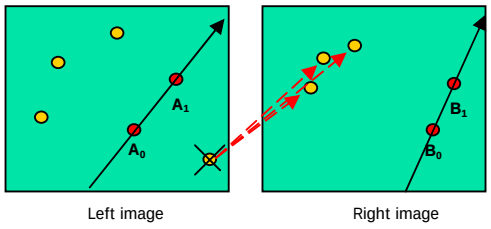

Figure 18: TTC – Step 4.2

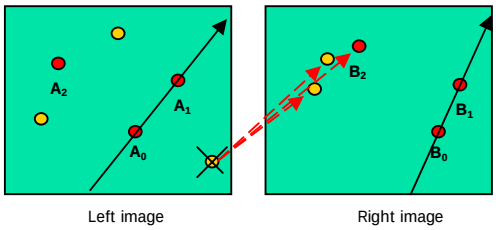

Figure 19: TTC – Step 4.3

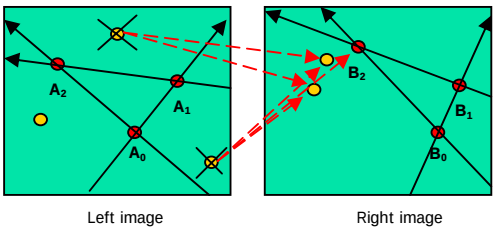

Figure 20: TTC - Step 4.2

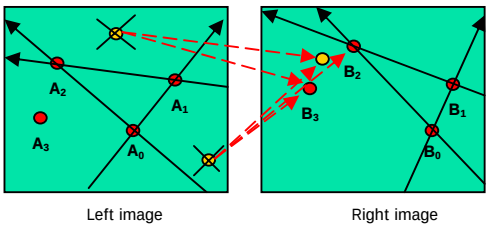

Figure 21: TTC - Step 4.3

5.3 True Tentative Correspondence tree estimation

To get the set of TTC proposed in the section 5.2 we have developed an algorithm which saves TTC in binary tree of the height 8, where every path from the root to the leaf is one TTC. The root represents correspondence from the Step 1, the first son of the root represents the first best different correpondence from the Step 2 and the second son the second best different correpondence from the Step 2. Then the i-th son of node at the level j represents the i-th best different correpondence from the Step 4.3. (i={1,2} j={2,3,4,5,6,7}).

The reason why we have decided to develop this algorithm is that the geometric correpondence on higher levels does not have to be the first best different correpondence but it can be the second one or higher. We can also simply compute n-ary tree of the height 8, but the binary tree takes less computation time and gives sufficient results.

5.4 Experiments

Left Image	Right Image	# inliers from first TTC	# inliers from TTC tree	# distinct TC from TTC tree	%
bookshA.tif	bookshB.tif	8	13	13	100%
castleA.tif	castleB.tif	8	12	12	100%
graffA.ppm	graffB.ppm	8	10	10	100%
kampaA.tif	kampaB.tif	8	10	11	91%
leafsA.jpg	leafsB.jpg	8	11	12	92%
plantA.tif	plantB.tif	5	8	15	53%
shout1.tif	shout3.tif	8	21	21	100%
vbnA.tif	vbnB.tif	8	9	9	100%
wallA.jpg	wallB.tif	3	9	21	43%
washA.tif	washB.tif	8	23	24	96%
chem_lab_01.png	chem_lab_02.png	8	10	10	100%
landhaus1.png	landhaus2.png	6	8	13	61%
landhaus2.png	landhaus3.png	8	10	12	83%
landhaus3.png	landhaus4.png	3	13	15	87%
mensa01.png	mensa02.png	8	11	11	100%
mensa02.png	mensa03.png	8	18	19	95%
Temmel01.png	temmel02.png	8	14	14	100%
Temmel02.png	temmel03.png	8	10	15	67%
Sum		**129:152(85%)**	**220**	**257**	**86%**

Table 3: Experiments results on TTC

Figure 22: Frames from vbnA.tif (left) and vbnB.tif (right) images and 8 correpondence regions from TCC method (blue).

5.5 Conclusions

In the section 5 the new TTC algorithm for finding tentative correpondences between MSER regions is presented and there is also experimentally shown, that in every case there exist at least 8 inliers in TTC tree and in 14 cases from all 18 experiments we have obtained geometric correspondences for all 8 correspondences from first TTC. As we have already mentioned, we need only 8 geometric correspondences to compute epipolar geometry between two images.

5.6 Future work

The essential idea of our future work is to traverse TTC tree and obtain set of TTC (every node of the set contains 8 tentative correpondences). Then for every TTC we assume that 8 correspondences are geometric and we verify this assumption by the use of epipolar geometry and some statistic methods.

6 Final conclusions

In the section 3 is described our implementation of MSER detection method and experimentally shown it's correctness.

The section 4 deals with establishing tentative correspondences between detected MSER regions and verify the method which uses LAF. We also propesed there our idea for changing the NCC method. Based on experiments the conculsion is that with the use of our Iterative NCC method we obtained better results as with original one, but these results are not satisfying for us.

Finally in the section 5 our TTC method to establish a set, where every node contains 8 tentative correspondences is described (wich are the best candidates to compute epipolar geometry). The results of experiments on TTC have been very satisfying. This is totally new approach to tentative correpondence estimation according to our opinion. This method provides a way how the slow stochastic RANSAC method can be substituted with use of the outputs from the deterministic TTC method.
This leads to a quick approximation of the WBS solution.

Appendix

A CD is attached to this thesis. Information about the CD content is written in the file "readme.txt" which is located in the root directory.

References

[1] *http://www.esat.kuleuven.ac.be/~pollefey/tutorial/node53.html*

[2] Frank A. van den Heuvel. *Towards automatic relative orientation for architectural photgrametry.* Dresden: International Archives of Photogrammetry and Remote Sensing, Vol. 34, Part 5, ISSN 1682 - 1750, pp. 227 – 232, 2002

[3] T. Tuytelaars, L. Van Gool. *Wide baseline stereo matching based on local, affinely invariant regions.*, Bristol : In Proc. 11th British Machine Vision Conference, pp. 412-425, 2000

[4] J. Matas, O. Chum, M. Urban, and T. Pajdla. *Robust wide baseline stereo from maximally stable extremal regions.* Image and Vision Computing 22, pp. 761-767, 2004

[5] D. I. Barnea, H. F. Silverman, *A class of algorithms for fast digital image registration.* IEEE Trans. Computers, 21, pp. 179-186, 1972

[6] J. Matas, O. Chum, M. Urban, and T. Pajdla. *Distinguished regions for wide-baseline stereo.* Prague: Center for Machine Perception, Research Report CTU–CMP–2001–33, K333 FEE Czech Technical University, November 2001

[7] Andre R., Hendriks E., Biemond J. *Correspondence Estimation in Image Pairs.* IEEE Signal Processing Magazine, pp. 29-46, May 1999

[8] Olivier Faugeras. *Three-Dimensional Computer Vision, A Geometric Viewpoint.* Boston: MIT Press, ISBN 0-262-06158-9, 1993

[9] CG Harris, M. Stephens, *A combined corner and edge detector.* In 4th Alvey Vision Conference, pp. 147-151, 1988

[10] K. Mikolajczyk and C. Schmid, *Scale and Affine invariant interest point detectors.* International Journal of Computer Vision 1(60): pp. 63-86, 2004

[11] DG Lowe. *Distinctive image features from scale-invariant keypoints.* International Journal of Computer Vision, 60(2), pp. 91–110, 2004

[12] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L. Van Gool. *A Comparison of Affine Region Detectors.* International Journal of Computer Vision. Submitted in August 2004

[13] O. Chum, J. Matas, *Randomized RANSAC with T. d,d. Test.* Bristol: British Machine Vision Conference, pp. 448-457, 2002

[14] V. Ferrari, T. Tuytelaars, L. Van Gool, *Wide-baseline multiple-view correspondences.* in Proc. IEEE. CVPR, pp. 718-725, 2003

[15] F. Fraundorfer, H. Bischof. *Evaluation of local detectors on non-planar scenes.* In Proc. 28th Workshop ÖAGM/AAPR 2004, pp. 125-132, 2004

[16] S. Obdrzalek and J. Matas. *Object recognition using local affine frames on distinguished regions.* Bristol: In Proc. British Machine Vision Conference, pp. 113-122, 2002

[17] S. Obdrzalek and J. Matas. *Local Affine Frames for Image Retrieval.* CIVR, pp. 318-327, 2002

[18] M.A. Fischler and R.C. Bolles. *Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography.* CACM, 24(6), pp. 381–395, June 1981.

[19] I. Kolingerová and A. Ferko. *Exploring Triangulations for Image Correspondence.* AMVH Technical report, 2003

[20] Matas, J. - Obdržálek, Š. - Chum, O.: *Local Affine Frames for Wide-Baseline Stereo.* In ICPR 02: Proceedings 16th International Conference on Pattern Recognition. Los Alamitos: IEEE Computer Society Press, pp. 363-366. ISBN 0-7695-1695-X, 2002