DOI:10.15774/PPKE.ITK.2011.005

Faculty of Information Technology at Péter Pázmány Catholic University

Laboratoire Bordelais de Recherche en Informatique at the University of Bordeaux 1

Video Event Detection and Visual Data Processing for Multimedia Applications

Dániel Szolgay

A thesis submitted for the degree of
Doctor of Philosophy

Supervisors:
Tamás Szirányi, D.Sc.
Jenny Benois-Pineau, D.Sc.

Scientific adviser:
Tamás Roska, D.Sc. ordinary member
of the Hungarian Academy of Sciences

Budapest, 2011

DOI:10.15774/PPKE.ITK.2011.005

DOI:10.15774/PPKE.ITK.2011.005

Acknowledgements

First of all I would like to thank my supervisors Professor Tamás Szirányi and Professor Jenny Benois-Pineau, for their consistent help and support in many ways and their guidance during my studies.

The advices, help, and encouragement of Prof. Tamás Roska are kindly acknowledged.

I thank all my colleagues whose ideas and advices assisted me during my work.

The support of the Péter Pázmány Catholic University and the University of Bordeaux 1, where I spent my Ph.D. years, is gratefully acknowledged. My studies in Bordeaux were financed by the French Government through "Bourses Eiffel" and "Bourses pour doctorat en cotutelle".

I am very grateful to my mother and father and to my whole family who always encouraged me during the long years and supported me in all possible ways.

DOI:10.15774/PPKE.ITK.2011.005

Abstract

This dissertation (i) describes an automatic procedure for estimating the stopping condition of non-regularized iterative deconvolution methods based on an orthogonality criterion of the estimated signal and its gradient at a given iteration; (ii) presents a decomposition method that splits the image into geometric (or cartoon) and texture parts using anisotropic diffusion with orthogonality based parameter estimation and stopping condition, utilizing the theory that the cartoon and the texture components of an image should be independent of each other; (iii) describes a method for moving foreground object extraction in sequences taken by wearable camera, with strong motion, where the camera motion compensated frame differencing is enhanced with a novel kernel-based estimation of the probability density function of the background pixels. The presented methods have been thoroughly tested and compared to other similar algorithms from the state-of-the-art.

DOI:10.15774/PPKE.ITK.2011.005

Contents

DOI:10.15774/PPKE.ITK.2011.005

DOI:10.15774/PPKE.ITK.2011.005

DOI:10.15774/PPKE.ITK.2011.005

DOI:10.15774/PPKE.ITK.2011.005

List of Figures

DOI:10.15774/PPKE.ITK.2011.005

DOI:10.15774/PPKE.ITK.2011.005

DOI:10.15774/PPKE.ITK.2011.005

DOI:10.15774/PPKE.ITK.2011.005

List of Tables

DOI:10.15774/PPKE.ITK.2011.005

DOI:10.15774/PPKE.ITK.2011.005

Summary

Problem Formulations

The efficiency of image and video analysis tasks are limited by physical factors: motion, motion blur, focusing error, edge detection problems because of shadows and textures, disparity problems. The first two parts of the dissertation address basic image enhancement problems such as optimizing deconvolution for image deblurring, and extraction of the geometrical structure of the image by decomposing it into texture and geometrical components, while in the third part, higher level video understanding will be examined, where the task is the detection of moving objects and their separation from a cluttered background in the video sequences recorded with a moving camera.

Image restoration is practically as old as image processing itself, constantly waiting for newer and better solutions. Deconvolution of blurred images, like the ones taken with strongly moving wearable cameras, gives a new motivation to solve an old challenge. Beside motion, there could be many other reasons of image blur like defocusing, atmospheric perturbations, optical aberrations. For these reasons, which are common in aerial, satellite or medical imaging, the acquired images are corrupted and restoration is needed. The distortion of the image is generally modeled as convolution: the original unknown image is convolved with a Point Spread Function (PSF) that describes the distortion. The goal of image processing here is obvious: restore the original image as well as possible based on the blurry measurement and the PSF. The usual approach is to look for an image that after the convolution with a known or estimated PSF, is most similar to the measured image. This approach leads to an ill-posed problem,since there is more than one image that would seem as a good solution. Hence it is a common drawback of non-regularized iterative deconvolution methods that after some iterations they start to am-

DOI:10.15774/PPKE.ITK.2011.005

plify noise. Our goal was to automatically find the optimal stopping point for these algorithms where the reconstructed image is as close to the unknown original image as possible.

The decomposition of an image into geometrical (cartoon) and noise like (texture) components is a fundamental task for both videos and still images. It can help image compression, denoising, image feature selection, or it can be a preprocessing step for video event detection: the same way as shadow, reflection or smoke/fog removal, the elimination of texture from the video frames aids the higher level understanding of the video. Theoretically the two parts are independent of each other: the cartoon image contains only geometrical information while the texture image, as complementary of the cartoon is free from geometrical information. A good algorithm may produce the cartoon image by removing all the texture from the original image without eliminating or blurring any cartoon part. The texture image then can be calculated as the difference of the original image and the cartoon.

Separating foreground objects from the background is a fundamental module for many video applications, as it is commonly used to bootstrap higher-level analysis algorithms, such as object-of-interest detection and tracking. The task is challenging for still camera recordings, but if wearable cameras are used, then strong motion and parallax, low quality of signal (reduced by motion blur) makes the problem even more complex. Deconvolution and cartoon/texture segmentation along with other low level algorithms can be used to enhance the results of this segmentation.

New Scientific Results

1. Thesis: *The stopping condition is a common problem for the non-regularized iterative deconvolution methods. a novel method has been introduced for automatically estimating the stopping condition based on the orthogonality of the change of the estimated signal between*

DOI:10.15774/PPKE.ITK.2011.005

two consecutive iterations and the signal itself. An effective lower bound estimate has been provided to the conventional ad-hoc methods and proved experimentally the efficiency of the proposed method for different noise models and a wide range of noise levels.

Finding the optimal stopping point for iterative deconvolution methods is an ill-posed problem. In a real life problem scenario only the acquired image and the PSF is available. In general, non-optimal *ad-hoc* methods are used to stop the iteration.

We have introduced a novel method for calculating the ideal stopping point for iterative non-regularized deconvolution processes, using the Angle Deviation Error (ADE) [1] measure instead of the commonly used Mean Square Error (MSE) measure.

The proposed method is capable of estimating the optimal stopping point of iterations based on the independence of an actual estimated signal and its gradient, indicating when an aimless section of the iterations is just starting, when the image is not enhanced anymore and only noise is added to it.

The proposed measure, $ADE(X_e(t), X(t))$ contains only measurable images and provides a reasonable solution for the stopping problem: at the minimum of $ADE(X_e(t), X(t))$ the change between two consecutive iterations $X_e(t)$ has the highest possible independence of the actual reconstructed image, hence we can assume that at this point $X_e(t)$ contains mostly independent noise and not structural information of the image, and further iteration will not enhance the image quality.

The method was tested with the well known Richardson-Lucy [2, 3] deconvolution algorithm with different noise models (Gaussian, Poisson) and wide range of noise levels. It does not require any input parameter or manual calibration. The correlation between the result of the theoretically best solution ($MSE(U, X(t))$) and the result of the proposed method ($ADE(X_e, X_re)$) is 0.6726. If we regard the

DOI:10.15774/PPKE.ITK.2011.005

correlation not in iteration number but in image quality, the value is even higher: 0.9556. We can conclude that the proposed method outperforms the generally used *ad-hoc* methods.

2. Thesis: *A novel axiomatic method has been proposed for the automatic separation of geometrical and textural components of the image. The heart of the algorithm is the Anisotropic Diffusion (AD), whose iteration is stopped adaptively to the image content, based on ADE orthogonality measure. It has been proved experimentally that the proposed method separates cartoon and texture components of the image with better quality than the recently published algorithms.*

The aim of the Anisotropic Diffusion [4] is to blur and filter the image from noise while it keeps the strong edges. For this it uses a weight function, which hinders the diffusion in the directions orthogonal to edges and allows it along the edges or in edge-free territories.

AD, as proposed in [4], is not suitable for cartoon/texture decomposition, since texture may contain high magnitude edges, which should be blurred and cartoon may contain weaker edges, which should be preserved.

The proposed algorithm utilizes cartoon image of the BLMV nonlinear filter [5] to initialize the weights for AD. In this image the textured regions are already blurred somewhat, hence the AD does not keeps them, while the main edges are preserved therefore the weighted inhibition of the AD will keep them untouched.

The iterative AD is stopped automatically based on the orthogonality of the two components using ADE measure. Our method was compared to the state-of-the-art methods of the field (TVL1 [6], ROF [7], DPCA [8], DOSV [9], AD [4] , BLMV [5]) using artificial images for numerical evaluation and real life images for visual comparison. The visual evaluation on real images is the most widely used method in spite of its subjectivity. Both evaluation approaches shows the proposed method superiority and contrary to the other algorithms it does

DOI:10.15774/PPKE.ITK.2011.005

not require precise manual tuning of the parameters, only a range of parameter values should be given as a starting condition.

3. Thesis: *Based on kernel density function estimation a novel method has been developed for moving foreground object extraction in sequences taken by a wearable camera (25fps, 320x240 frame size), with strong and unpredictable motion.*

Foreground extraction on wearable camera recordings is a challenging task since the camera motion is unpredictable and strong, and motion blur and intensive noise corrupt the quality of images.

Working with moving cameras the estimation and compensation of the camera motion is the first step towards moving foreground detection. We have performed a Hierarchical Block-Matching (HBM) [10] and affine Global Motion Estimation (GME) [11] to compensate camera motion.

After this step two consecutive frames of the video can be represented in the same coordinate system and the error image can be calculated as the absolute difference of the two frames. This error image should contain high values only on the pixels corresponding to moving foreground objects, while the static background points should have low values after the difference calculation. Due to changes of the perspective, quantization errors and errors of the motion compensation the error image cannot be used as foreground model, because of the large number of false positives. A Modified Error Image (MEI) has been formed, which contains the color information of the original frame and the motion information of the error image.

To separate pixels of moving objects from pixels in static contours present in MEI due to the noise, Probability Density Function (PDF) estimation of the background and probabilistic decision rule is used.

The estimation of the PDF was done based on samples from a spatial-temporal patch with kernel density estimation [12], using Gaussian kernel. It is called spatiotemporal according to the choice of sample

DOI:10.15774/PPKE.ITK.2011.005

points: spatial neighborhood and temporal history of a pixel are both used.

For the bandwidth calculation we propose to use the distance from all the k nearest neighbors, instead of the distance from the kth alone, since the latter may give us false result when the number of sample points is strongly limited. In the given circumstances (low number of sample point, strong noise) the sample point selection technique has key importance.

A common approach for selecting the sample points in case of still cameras for a given (x, y) coordinate is to use the n previous measurements taken at the same (x, y) position [13]. When the camera is moving the case is different. Even after motion compensation the real background scene position that corresponds to the (x, y) pixel in one frame, might move a little, due to errors of camera motion compensation, or quantization. Assuming that this spatial error is random, the values selected in a small patch centered on the pixel (x, y) are used. Based on the values of the M measurement matrix, which contains the last n motion compensated frames, a joint PDF is built for the color channels of each non zero pixel of the current MEI.

Once the PDF has been built for each pixel in the current frame, we can proceed to the detection of foreground moving objects. Here the pixels will be first classified as foreground or background based on an adaptive threshold that considers the PDFs characteristics. Then the detected pixels will be grouped into clusters (moving objects) with DBSCAN algorithm [14] on the basis of their motion, color and spatial coordinates in the image plane.

It has been proved in an experimental way that the proposed framework gives better detection results than the widely used Stauffer-Grimmson method [15]. The calculations are done in offline mode at the moment, since the computational cost is too high for real-time processing.

DOI:10.15774/PPKE.ITK.2011.005

Examples for Application

Wearable video monitoring has a lot of potentials in the fields of health care, security and social life. It can be an important tool for diagnosing aged dementia, where the traditional methods may fail, since the patients cannot or voluntarily will not help the physicians to diagnose their disease. Using video logs about the life of the patients can help the doctors in their work. For security surveillance it can be an effective tool using together with static cameras or in cases when the use of static cameras is not an option (e.g. police patrols).

Blogging and life logging is becoming more and more popular. The author writes down his or her life like in a diary, but using the possibilities of the electronic world, uploading pictures, videos and music. A research project of Microsoft, called SenseCam [16] is helping the users to build a diary with photos using a special wearable camera that documents the users whole day with pictures. (This is a way of modern entertainment, but also it could be used in health care curing patients with memory disorders.) With wearable cameras and the necessary handling algorithms video diaries would also be available for the blogging society.

Our work in separation of the foreground and the background is just the first step toward content based search of videos, which is one of the most intensively researched areas of multimedia and computer vision.

The decomposition of an image into cartoon and texture components could be a starting point for many algorithms. It could be useful for image compression where compressing the cartoon and the texture components separately can provide better results [17]. Such a coding proved to be efficient in the past [18, 19]. It is applicable for image denoising [7] since zero mean oscillatory noise can be regarded as a fine texture, image feature selection [6] and main edge detection as illustrated in [5] etc. In motion estimation it could be used to eliminate the effect of noise, which may reduce the precision of the

DOI:10.15774/PPKE.ITK.2011.005

estimation.

Deconvolutional methods are widely used in image processing where defocusing is an issue: from microscopy to astronomy. It could be a preprocessing step for videos taken by wearable cameras, where motion blur corrupts the frames. Although nowadays regularization is the main trend, non-regularized methods are also capable producing results comparable to the state-of-the-art [20]. For non-regularized methods the stopping problem is a key issue. The method we were working on offers a logically sound and effective solution to this problem.

DOI:10.15774/PPKE.ITK.2011.005

Magyar Nyelvű Összefoglaló

A Problémák Ismertetése

A képek és videók feldolgozásának hatékonyságát nagyban befolyásolhatják fizikai tényezők, mint a mozgás, mozgásból adódó elmosódás, fókuszálási hiba, árnyékok/textúrák jelenlétéből adódó élmeghatározási hibák, diszparitási problémák. A disszertáció első két részében alap képjavítási problémákkal foglalkoztam úgymint a képek elmosódásának megszüntetését segítő dekonvolúciós algoritmusok optimalizálása illetve a kép pusztán geometriai információt tartalmazó részének előállítása a textúra elkülönítésével, míg a harmadik rész egy magasabb rendű videó értelemezési feladatra koncentrál, melynek célja mozgó objektumok elkülönítése a statikus háttértől mozgó kamerával készült felvételeken.

A kép elmosódottságnak számos oka lehetséges, mint például hibásan beállított fókusz távolság, gyorsan mogó kamera, a felvevő optika hibája. Az elmosódottság okozta torzulást általában konvolúcióval szokták modellezni: az eredeti ismeretlen képet konvolváljuk egy pontszóródási függvénnyel (PSF). A PSF egy pontszerű fényforrás a képalkotás során elszenvedett torzulását írja le. A cél egyértelmű: a lehető legjobb minőségben visszaállítani az eredeti képet az elmosódott kép és - bizonyos esetekben - a PSF alapján. A legtöbb eljárás úgy próbálja megoldani a problémát, hogy keresi azt a képet, mely konvolválva a becsült (vagy pontosan ismert) PSF-el a lehető leghasonlóbb lesz a mért, elmosódott képhez. Ez a megközelítés azonban alulhatározott problémát eredményez, mivel a keresett eredeti kép mellett sok más kép is kielégíti a fenti feltételt. Ennek hatására sok nem-regularizált iteratív dekonvolúciós módszer közös gyengéje, hogy előbb-utóbb zajt visznek a becsült képre. A céunk az volt, hogy találjunk olyan automatikusan számolható megállási feltételt, mely az iteratív folyamatot a legoptimálisabb pontban állítja le.

DOI:10.15774/PPKE.ITK.2011.005

Videók és álló képek esetén egyaránt fontos feladat a képek felbontása geometriai és zajszerű komponensekre. A két rész elméletileg független egymástól: az ún. cartoon kép csak geometriai információt tartalmaz, míg a textúra kép, a cartoon rész komplementereként áll elő és nem tartalmaz geometriai információt. Egy jó dekompozíciós algoritmus eltávolítja a textúrát a képről anélkül, hogy elmosná a fontos körvonalakat. A textúra kép ezután az eredeti és a cartoon kép különbségeként állítható elő.

Az előtér objektumok háttértől való elszeparálása egy olyan alapvető feladat, mely nagy érdeklődésre tarthat számot, hiszen ennek eredménye számos magasabb szintű algoritmus (pl. objektumok detektálása és követése) kiinduló pontja lehet. Az erős kameramozgás, a jelentős perspektíva változás és a felvételek zajossága még jobban megnehezíti a feladatot testen viselhető kamerák esetén. Dekonvolúciós módszerek, cartoon/textúra szeparálás és más alacsony szintű algoritmusok segíthetnek a lehető legjobb eredmény elérésében.

Új Tudományos Eredmények

1. Tézis: *A megállási feltétel meghatározása általános probléma a nem regularizált iteratív dekonvolúciós módszerek esetében. Új módszert adtam az ideális megállási pont automatikus meghatározásához, a mért jel és a jel gradiensének ortogonalitása alapján. A módszer alkalmas az eddig használt ad-hoc eljárások négyzetes hibájának alsó burkolót adni. Az elméleti megfontolást kísérletekkel támasztottam alá, melyek bizonyítják az algoritmus hatékonyságát különböző zaj modellek és jel-zaj viszony esetén.*

Új módszert dolgoztunk ki az ideális megállási pont automatikus meghatározásához nem regularizált iteratív dekonvolúciós módszerek esetén az ADE [1] ortogonalitás mértéket használva a széles körben használt négyzetes hiba mérték (MSE) helyett.

DOI:10.15774/PPKE.ITK.2011.005

A javasolt módszer alkalmas az iteráció optimális pontban való leállítására az aktuálisan becsült jel és a jel gradiensének függetlensége alapján, megelőzve ezzel az iterációknak egy olyan szakaszát, amely nem javítja tovább a képet, csak zajt ad hozzá. A javasolt $ADE(X_e(t), X(t)$függvény csak mérhető értékeket tartalmaz, vagyis minden adat rendelkezésre áll a kiszámításához és elméletileg is értelmezhető. Az $ADE(X_e(t), X(t)$ függvény minimumánál a két egymást követő iterációban tett becslés közti különbség $X_e(t)$ a lehető legfüggetlenebb magától a becsült képtől ezért feltételezhetjük, hogy $X_e(t)$ nagyrészt független, zajszerű információt tartalmaz és nem a kép struktúrájára vonatkozó információt. Ezért a további iterálás inkább rontja, mint javítja a kép minőségét.

A módszerünket a széles körben ismert Richardson-Lucy [2, 3] dekonvolúciós algoritmus használatával teszteltük különböző zaj modellekkel (Gaussi és Poisson) és eltérő zaj szinttel. Az eljárás nem igényel semmilyen kalibrációt vagy manuális beállítást. A javasolt módszer $(ADE(X_e, X_re))$ és az elméletileg legjobb megoldás $(MSE(U, X(t)))$ közti korreláció 0.6726, ha az iteráció számot vesszük alapul. Míg ha a kép minőségét tekintjük, akkor a korreláció még magasabb 0.9556 lesz. A disszertációban bemutatott eredmények alapján elmondható, hogy a javasolt módszer egyértelműen jobban teljesít, mint az általában használt *ad-hoc* eljárások.

2. Tézis: *Új, axiomatikus módszert adtam a képen szereplő geometriai és textúra részek automatikus szétválasztására. Az eljárás alapját anizotrop diffúzió adja, melynek képtartalomtól függő, megfelelő iterációban történő leállításához az ADE ortogonalitás mértéket használtam. Kísérletekkel bizonyítottam, hogy a létrehozott új eljárás jobb eredménnyel választja szét a képen a textúrát és a geometriai információt, mint az utóbbi években publikált módszerek.*

Az Anizotrop Diffúzió (AD) [4] célja, hogy a képen úgy hajtson végre elmosást és ez által zajszűrést, hogy a képen szereplő erősebb éleket érintetlenül hagyja. Ehhez az összes lehetséges diffúziós irányban egy

DOI:10.15774/PPKE.ITK.2011.005

súlyfüggvényt használ, ami meggátolja a diffúziót az adott irányba, ha ott az irányra merőleges él szerepel és megengedi a diffúziót, ha nincs ilyen él. Az AD hagyományos formájában nem alkalmas a geometriai (más néven cartoon) és a textúra információ szétválasztására, mivel a textúra is tartalmazhat erős éleket, amiket el kellene mosni, míg a cartoon is tartalmazhat gyenge éleket, amiket meg kéne őrizni. A javasolt eljárás a BLMV nem lineáris szűrő [5] által készített cartoon képet használja az AD súlyfüggvényének inicializálásához. Ezen a képen a textúrált részek már bizonyos mértékben el vannak mosva, így az AD nem fogja megőrizni őket, míg a fontosabb cartoon élek megmaradnak, így az AD súlyfüggvénye meg fogja védeni őket az elmosódástól. Az iteratív AD-t a két komponens közti ortogonalitási feltételt felhasználva, az ADE mérték segítségével automatikusan állítjuk le.

Az elkészült algoritmust összehasonlítottuk a ma elérhető legjobb hasonló módszerekkel (TVL1 [6], ROF [7], DPCA [8], DOSV [9], AD [4] , BLMV [5]), mind mesterséges képeken numerikus kiértékelést alkalmazva, mind valós felvételeken jól meghatározott szempontokat alapján. Az eredmények valós képeken történő értékelése, nyilvánvaló szubjektivitása ellenére a ma használt legelterjedtebb módszer.

Mindkét kiértékelési megközelítés az itt bemutatott módszer egyértelmű fölényét mutatja. A javasolt módszer további előnye, hogy a többivel ellentétben nem igényel pontos manuális paraméterezést, csupán egy paraméter tartomány megadása szükséges.

3. Tézis: *Kernel sűrűségfüggvény becslésen alapuló új eljárást dolgoztam ki mozgó előtér detektálására viselhető kamerával készült felvételekhez (25 fps, 320x240 képméret), melyeket általában erős és kiszámíthatatlan kameramozgás jellemez.*

Mozgó kamerával készült felvételek feldolgozása esetén a kamera mozgás becslése és kompenzálása az első lépés, melyet hierarchikus blokk-illesztő algoritmus (a továbbiakban HBM) [10] és affin globális

DOI:10.15774/PPKE.ITK.2011.005

mozgást becslő eljárás (GME) [11] felhasználásával valósítottunk meg.

Ezáltal a videó egymást követő két képkockája ábrázolhatóvá válik egy közös koordináta rendszerben és a hibakép előáll a két kép abszolút különbségeként. Ez a hibakép ideális esetben csak előtérpontokban tartalmazna magas értékeket, míg a statikus háttér pontok a különbség képzés során kioltanák egymást. A perspektíva megváltozása, kvantálási hiba és a mozgáskompenzáció kisebb pontatlansága következtében sok a hibás pozitív találat, ezért a hibakép önmagában nem alkalmas előtér modellnek. Létrehoztunk egy módosított hibaképet (MEI), ami a mozgáskompenzált különbségkép kiegészítve az aktuális képkockáról származó szín információval.

A mozgó objektumok és a hiba képen jelenlévő statikus háttér elemek pixeleinek elkülönítéséhez a háttér sűrűségfüggvényének a becslését és egy valószínűség alapú döntési szabályt dolgoztunk ki.

A sűrűségfüggvény becslését tér-időbeli ablakból vett minták alapján kernel sűrűség becslés [12] segítségével végeztük, Gaussi kernelt alkalmazva. A tér-időbeliség arra utal, hogy a minta pontokat egy térbeli környezet különböző időpillanatokban vett értékeiből választottuk.

A kernel függvény szélességének beállításához a k legközelebbi minta pontot vettük figyelembe ahelyett, hogy csak a k. pontot használtuk volna, így csökkentve az alacsony minta számból fakadó esetleges hibákat.

A minta pontok száma a jelen feladatban erősen korlátozott és a zaj esetenként nagyon erős lehet, ezért a mintapontok választásának módja kulcsfontosságú.

Közismert eljárás a mintapont választásra rögzített kamerák esetén egy adott (x, y) koordinátájú pixel n korábbi értékének használata [13]. Mozgó kamera esetén azonban ez a módszer nem megbízható. A mozgás kompenzáció ellenére egy valós, statikus háttérpont, ami egy adott képkockán az (x, y) koordinátájú pontnak felel meg a következő képen lehet, hogy nem pont ugyanott lesz. Ez magyarázható a mozgás

DOI:10.15774/PPKE.ITK.2011.005

kompenzáció kisebb hibáival vagy kvantálásból adódó hibával egyaránt. Ezt a hibát térben véletlenszerűnek feltételezve egy kis (x, y) középpontú térbeli ablak használatát javasoltuk.

Ezt követően az M mátrix értékei alapján egy együttes valószínűségi sűrűség függvényt számoltunk a színcsatornákra a MEI minden nem nullaértékű pontjára. Ahol az M mérési mátrix mindig az n megelőző, mozgáskompenzált képkockát tartalmazza.

Az így kapott sűrűségfüggvények alapján minden pixelt előtérnek vagy háttérnek osztályoztunk egy, a függvények karakterisztikáját figyelembe vevő adaptív küszöbölés segítségével. A kapott előtér pontokat újra klasszifikáltuk a hozzájuk tartozó mozgás koordináták, szín értékek és térkoordináták alapján a DBSCAN klaszterező algoritmus segítségével.

Kísérleti úton bebizonyítottuk, hogy a bemutatott eljárás hatékonyabban működik viselhető kamerával készített felvételek esetén, mint a jól ismert Stauffer-Grimmson [15] módszer. Jelenleg az algoritmus offline működésre képes, mivel a nagy számítási igénye nem teszi lehetővé a valós idejű futtatást.

Az Új Tudományos Eredmények Lehetséges Felhasználási Területei

A viselhető kamerákkal készített videó megfigyelés rengeteg lehetőséget hordoz magában az egészségügyi, biztonság technikai vagy akár a szociális élet területén. Fontos kiegészítő eszköze lehet az időskori demencia diagnosztizálásának olyan esetekben, amikor a hagyományos módszerek sikertelenek, mivel a páciensek nem tudják, vagy nem akarják segíteni az orvosok munkáját. Videó logok készítésével az orvosok betekintést nyerhetnek a beteg mindennapjaiba, ami adott esetben nagy segítség lehet a helyes kórkép felállításához. Biztonsági megfigyeléskehez is fontos eszköz lehet a

DOI:10.15774/PPKE.ITK.2011.005

viselhető kamera olyan körülmények között, amikor a hagyományos statikus kamerák használata nem lehetséges (pl. rendőr járőrökön).

Manapság egyre népszerűbbek a blogok és az ún. life logok, melyekben a szerzőjük saját életét írja le nagyjából úgy, mint egy naplóban, kiegészítve a modern technika adta lehetőségekkel (képek, zenék, videók felöltésével). A Microsoft SenseCam [16] projektje egy viselhető kamerával a hordozója egész napját fényképekkel dokumentálja, lehetővé téve egy fényképekből álló napló könnyű létrehozását. (Ennek a szórakoztatás mellett orvosi felhasználásai is lehetnek memória zavarral küzdő betegek esetén.) Viselhető videó kamerákkal és megfelelő feldolgozó algoritmusokkal a SenseCamhez hasonló videó naplók készítése is vélhetően vonzó lenne a blog író társadalom számára.

A geometriai és textúra információ szétválasztása sok egyéb algoritmus számára jelenthet jó kiinduló pontot. Tömörítési eljárásoknál a két komponens külön választásával jobb eredmény érhető el [17], ahogy azt korábbi módszerek megmutatták [18, 19]. Él kereső eljárásoknál a fontos élek megtalálásához adhat segítséget [5], képi jellemzők kinyerésére [6], valamint zajszűrésre is alkalmas abban az esetben, ha nulla középértékű véletlen zajjal van terhelve a kép [7]. Kamera mozgás becslésénél is hasznos lehet, a becslést hátráltató zaj hatásának csökkentésére.

Dekonvolúciós módszerek használata mindennapos olyan területeken, ahol digitális képeket alkalmaznak és az elmosódottság problémát jelenthet (pl.: mozgó kamerás felvételek, mikroszkópia, asztronómia). Bár manapság a regularizáció számít a fő irányvonalnak a területen, a nem regularizált módszerekkel is minőségi eredményeket lehet elérni [20]. Ezeknél a módszereknél az iterációt megállító feltétel kulcskérdés. A kidolgozott módszer elméletileg értelmezhető és effektív megoldást kínál a problémára.

DOI:10.15774/PPKE.ITK.2011.005

Résumé en Français

Formulation des Problèmes

Cette thèse de doctorat est consacrée aux problèmes de traitement d'images et d'analyse vidéo, problèmes qui restent ouverts et qui sont rencontrés dans un vaste éventail d'applications du domaine multimédia. La première partie de la thèse est consacrée à la restauration d'images floues, en particulier par déconvolution. Il y a beaucoup de raisons pour lesquelles une image, acquise avec un appareil photo ou une caméra vidéo, peut être floue: défocalisation, mouvements rapides de la caméra, perturbations atmosphériques ou aberrations optiques. Ce type de distorsion est généralement modélisé comme une convolution: l'image d'origine inconnue est convoluée avec une fonction modélisant une réponse impulsionnelle du système d'acquisition engendrant le flou (Point Spread Function en anglais, PSF) et décrivant la distorsion. L'objectif est évident: restaurer l'image originale aussi bien que possible en se basant sur la mesure du flou engendré par la PSF.

L'approche habituelle consiste à comparer une image après convolution par une PSF connue ou estimée avec l'image observée. Ainsi, en minimisant une fonctionnelle de l'erreur de mesure de façon itérative, la PSF peut être estimée et l'image déconvoluée. Cette approche conduit à un problème mal posé: la solution n'en est pas unique. L'inconvénient majeur des méthodes itératives de déconvolution est que tôt ou tard dans l'itération, elles commencent à amplifier le bruit. Notre objectif est de trouver automatiquement le point d'arrêt optimal pour ces algorithmes, point où l'image reconstruite est aussi proche que possible de l'image inconnue originale. Nous avons donc proposé une solution à ce problème.

Dans la deuxième partie de ce travail de thèse nous nous sommes appliqués à décomposer les images en composantes "plates", "cartoon"

DOI:10.15774/PPKE.ITK.2011.005

et texture. La décomposition d'une image en éléments géométriques (cartoon) et bruit (texture) est une tâche fondamentale pour les vidéos et pour les images fixes. Théoriquement les deux parties sont indépendantes les unes des autres: l'image de cartoon ne contient que des informations géométriques tandis que la texture de l'image, peut être considérée comme complémentaire de la partie "cartoon". Un bon algorithme doit produire l'image de "cartoon" en enlevant toute la texture de l'image originale sans éliminer ou bruiter la composante "plate". La texture de l'image peut ensuite être calculée comme la différence entre l'image originale et la composante "cartoon".

Dans la troisième partie de la thèse nous nous intéressons à un autre problème de séparation. Cette fois il s'agit de séparer les objets d'avant-plan dans des séquences vidéo du fond de la scène. La séparation des objets de premier plan par rapport à l'arrière-plan est un module fondamental pour de nombreuses applications vidéo et il est communément utilisé pour aider des algorithmes de niveau supérieur, tels que la détection d'objets d'intérêt ou le suivi d'objets. Cette tâche est difficile pour des vidéos acquises avec des caméras fixes, mais si des caméras portables sont utilisées, le mouvement fort de la camera, la parallaxe et la faible qualité du signal rend le problème encore plus complexe. Ce dernier travail s'est déroulé dans un contexte de recherche pluridisciplinaire d'utilisation de caméras portées par des patients pour les objectifs d'études épidémiologiques et de diagnostic des démences liées à l'âge. La déconvolution de notre première partie, la séparation "cartoon"/texture de notre deuxième partie ainsi que d'autres algorithmes de bas niveau peuvent être utilisés pour améliorer les résultats de cette segmentation.

Nouveaux Résultats Scientifiques

1. Thèse: *La définition de la condition d'arrêt est un problème commun pour les méthodes de déconvolution itérative sans régularisation.*

DOI:10.15774/PPKE.ITK.2011.005

Une nouvelle méthode a été introduite pour estimer automatiquement l'état d'arrêt. Cette méthode est basée sur l'orthogonalité du changement du signal estimé entre deux itérations consécutives et le signal lui-même. Une limite efficace inférieure de cette estimation a été fournie aux méthodes classiques ad hoc et a montré expérimentalement l'efficacité de la méthode proposée pour des modèles de bruit différents et un large éventail de niveaux de bruit.

Trouver le point d'arrêt optimal pour les méthodes de déconvolution itératives est un problème mal posé dans un scénario réel quand l'image acquise et la PSF sont disponibles. Classiquement, des méthodes ad hoc non optimales sont utilisées pour arrêter l'itération. Nous avons introduit une nouvelle méthode de calcul de la condition d'arrêt idéale pour les processus itératifs de déconvolution sans régularisation en utilisant la mesure appelée l'erreur angulaire de déviation (ADE) [1] au lieu de la mesure classique d'erreur quadratique moyenne "Mean Square Error" (MSE). La méthode proposée est capable d'estimer le point d'arrêt optimal d'itérations en se basant sur l'indépendance d'un signal réel estimé et sur son gradient. Elle indique également le moment où une série d'itérations devient "inutile", c'est à dire qu'au cours de ces itérations, l'image ne sera pas améliorée, mais qu'au contraire une amplification du bruit commence.

L'équation de calcul de cette mesure ne comprend que des informations issues d'images mesurables et fournit une solution raisonnable au problème d'arrêt: au minimum de $ADE(X_e(t), X(t))$ la variation entre deux itérations consécutives $(X_e(t))$ a la plus grande indépendance possible par rapport à l'image réelle reconstruite. Il est donc possible de supposer qu'à ce stade le gradient $X_e(t)$ (cf. éq. (3.6)) contient majoritairement du bruit indépendant et non pas des informations structurelles de l'image. Par conséquent, les itérations suivantes n'amélioreront pas la qualité d'image reconstruite. Cette méthode a été testée avec l'algorithme de déconvolution de Richardson-Lucy [2, 3] avec des modèles différents de bruit (Gaussien, Poisson)

DOI:10.15774/PPKE.ITK.2011.005

et un large éventail de niveaux de bruit. Elle ne nécessite aucun paramètre d'entrée ou d'étalonnage manuel.

La corrélation entre la meilleure solution théorique à la base de $(MSE(U, X(t)))$ et du critère proposé $(ADE(X_e, X_re))$ est assez forte (la valeur du coefficient de corrélation en nombre d'itérations s'élève à 0,6726). Si l'on considère la corrélation non pas en nombre d'itérations, mais en terme de qualité d'image, la valeur est encore plus élevée (0,9556). Nous pouvons conclure que la méthode proposée surpasse les méthodes généralement utilisées *ad-hoc*.

2. Thèse: *Une nouvelle méthode axiomatique a été proposée pour la séparation automatique des composantes géométriques et de texture de l'image.*

Le coeur de l'algorithme est la diffusion anisotrope (DA), dont l'itération est arrêtée de manière adaptative est basé sur la mesure d'orthogonalité ADE introduite précédemment. Il a été montré expérimentalement que la méthode proposée sépare les composants "cartoon" et "texture" de l'image avec une meilleure qualité que les algorithmes récemment publiés.

L'objectif de la diffusion anisotrope [4] est de lisser et de filtrer l'image tout en préservant les contours forts. Pour cela, une fonction de poids est utilisée. Cette fonction empêche la diffusion dans les directions orthogonales aux contours, et l'autorise le long des contours ou dans des zones sans contours.

DA, dans sa forme originale, n'est pas adaptée à la décomposition "cartoon"/texture puisque la composante de texture peut présenter des contours de grande amplitude, qui doivent être lissés, et la composante "cartoon" peut contenir des contours plus faibles, qui doivent être préservés. L'algorithme proposé utilise l'image "cartoon" obtenue par un filtrage non-linéaire [5] pour initialiser le poids de la DA. Dans cette image, les régions texturées sont déjà légèrement lissées. Donc la DA ne les conserve pas tandis que les contours principaux sont

DOI:10.15774/PPKE.ITK.2011.005

conservés. Ainsi, l'inhibition pondérée de la DA les garde intacte.

Le processus itératif de la DA est arrêté automatiquement en se basant de l'orthogonalité des deux composantes en utilisant la mesure ADE. L'algorithme proposé a été comparé aux méthodes de l'état de l'art du domaine (TVL1 [6], ROF [7], DPCA [8], DOSV [9], DA [4] , BLMV [5]) utilisant des images artificielles pour l'évaluation numérique et des images réelles pour une comparaison visuelle. Il est à noter que l'évaluation visuelle sur des images réelles est la méthode la plus largement utilisée en dépit de sa subjectivité.

Les deux approches d'évaluation montrent la supériorité de la méthode proposée en termes de qualité. Par ailleurs, et contrairement aux autres méthodes, elle ne nécessite pas de réglage manuel précis des paramètres, seule une gamme de valeurs de paramètres doit être proposée comme condition de départ.

3. Thèse: *En se basant sur l'estimation de la fonction de densité à noyaux, un nouveau procédé a été développé pour l'extraction d'objets en mouvement d'avant-plan dans des séquences vidéo acquises par une caméra portée par des personnes (25fps pour une taille d'image de 320x240), avec un mouvement fort et imprédictible.*

L'extraction des objets d'avant-plan dans des séquences vidéo des caméras portées (mobiles) est une tâche difficile puisque le mouvement de la caméra est fort et le flou de mouvement et le bruit corrompent la qualité des images vidéo acquises. Dans le cadre des séquences acquises avec des caméras en mouvement, l'estimation et la compensation du mouvement de la caméra est le premier pas vers la détection des objets d'avant plan en mouvement propre. Vu la forte amplitude du mouvement de la caméra, nous avons effectué une compensation par la mise en correspondances hiérarchique des blocs (HBM) [10] et nous avons ensuite utilisé le champs éparse de déplacements pour l'estimation du modèle affine du mouvement global (GME) [11] pour compenser le mouvement de la caméra.

DOI:10.15774/PPKE.ITK.2011.005

Après cette étape, deux images consécutives de la vidéo peuvent être représentées dans le même repère et l'image d'erreur peut être calculée comme la différence absolue des deux images. Cette image d'erreur doit contenir des valeurs élevées uniquement sur les pixels correspondant aux objets de premier plan avec l'ego-motion, tandis que les points de fond statique doivent avoir des valeurs faibles. En raison des changements de point de vue, des erreurs de quantification et des erreurs de compensation de mouvement, cette image d'erreur, telle quelle, ne peut pas être utilisée comme modèle de premier plan. Elle contient en effet un nombre élevé de faux positifs. Nous avons donc calculé une image d'erreur modifiée (MEI), qui contient les informations de couleur de l'image originale et les informations de mouvement de l'image d'erreur.

Pour séparer les pixels appartenant aux objets en mouvement des pixels de contours statiques présents dans la MEI en raison du bruit, nous avons proposé l'estimation de la fonction de densité de probabilité (PDF) de l'arrière plan ainsi qu'une règle de décision probabiliste quant à l'appartenance des pixels aux objets en mouvement ou aux contours statiques "mal compensés".

Pour chaque pixel d'amplitude non nulle de l'image MEI, l'estimation de la PDF a été réalisée sur des échantillons d'un patch spatio-temporel autour du pixel. Cette estimation a été réalisée par l'approche à noyaux [12], en utilisant un noyau gaussien. Nous appelons la méthode et la PDF estimée "spatio-temporelles" selon le choix des points-échantillons: le voisinage spatial d'un tel patch, et l'historique temporel d'un pixel sont tous deux utilisés. Pour le calcul de la bande passante du noyau dans l'esprit kppv nous proposons d'utiliser la distance de tous les k plus proches voisins, au lieu de la distance du k_i-ème seul, puisque celle-ci peut nous donner de faux résultats lorsque le nombre de points échantillons est limité. Or dans les circonstances données (nombre réduit de d'échantillons, bruit fort) la méthode de sélection des points-échantillons a une importance ma-

DOI:10.15774/PPKE.ITK.2011.005

jeure.

Une approche commune de sélection des points-échantillons dans le cas de caméras fixes pour une coordonnée donnée (x, y) est d'utiliser les n mesures antérieures prises à la même position [13]. Si la caméra est en mouvement, le cas est différent. En effet, même après la compensation de mouvement, la position réelle des pixels du fond de la scène animée correspondant au pixel (x, y) dans une image peut être erronée en raison d'erreurs de compensation de mouvement de la caméra ou bien de la quantification. En supposant que cette erreur spatiale est aléatoire, les valeurs sélectionnées dans un petit patch centré sur le pixel (x, y) sont utilisées.

Sur la base des valeurs de la matrice de mesure M, qui contient les n dernières images compensées en mouvement, la fonction de densité de probabilité (PDF) associée est estimée pour les chaînes de valeur de mesures couleur de chaque pixel non nul de l'image MEI à l'instant de temps courant.

Une fois les PDF estimées, nous procédons à la détection des objets en mouvement. Les pixels sont d'abord classés comme appartenant à l'avant-plan ou à l'arrière-plan sur la base d'un seuil probabiliste qui tient compte des caractéristiques des PDF. Ensuite, les pixels détectés sont regroupés en classes (objets en mouvement) avec l'algorithme de classification non-supervisée DBSCAN [14]. Le vecteur-mesure contient ici les caractéristiques du mouvement du pixel, sa couleur et ses coordonnées spatiales dans le plan d'image.

Nous avons montré de façon expérimentale que le cadre proposé donne de meilleurs résultats de détection des pixels d'avant-plan que la méthode de Stauffer et Grimmson [15] appliquée aux séquences compensées en mouvement. Les calculs ont été effectués en mode hors ligne pour le moment, puisque le coût de calcul est encore trop élevé pour le traitement en temps réel.

DOI:10.15774/PPKE.ITK.2011.005

Exemples d'Application

La surveillance vidéo avec caméras portées a beaucoup de potentiel dans les domaines de la santé, de la sécurité et de la vie sociale. Elle peut être un instrument important pour le diagnostic des démences liées à l'âge à partir des données de l'observation vidéo rapprochée, lorsque les méthodes traditionnelles peuvent échouer, car les patients ne peuvent pas aider les médecins à diagnostiquer la maladie. L'utilisation des "journaux vidéo" de la vie des patients peut ainsi aider les médecins dans leur travail.

Pour la vidéo-surveillance dans le domaine de la sécurité, l'analyse de la vidéo en provenance de caméras mobile peut être un outil efficace en l'utilisant conjointement avec des caméras statiques ou dans les cas où l'utilisation de caméras statiques n'est pas une option (par exemple les patrouilles de police).

Le blogging quant à lui est devenu de plus en plus populaire, les auteurs décrivant leur vies comme dans un journal, mais en utilisant maintenant les possibilités du monde numérique comme le téléchargement de photos, de vidéos et de musique. SenseCam [16], projet de recherche de Microsoft, consiste à aider les utilisateurs à construire un journal avec des photos prises lors de la journée entière à l'aide d'une caméra spéciale portable (il s'agit d'un moyen de divertissement moderne, mais il pourrait être aussi utilisé dans les soins de santé pour les patients avec des troubles de la mémoire). Avec le développement des appareils portables et des algorithmes de traitement associés, les "journaux vidéo" pourront être également présents dans le cadre des réseaux sociaux . L'extraction des objets en avant plan animés par le mouvement propre est une étape nécessaire pour l'anonymisation des données et le respect de la vie privée, pour la génération des alarmes suite à l'apparition d'objets d'intérêt, etc.

Ainsi l'ensemble des méthodes développées dans ce travail de thèse pourront trouver une application immédiate.

DOI:10.15774/PPKE.ITK.2011.005

La décomposition d'une image en éléments constitutifs "cartoons" et texture pourrait être un point de départ pour de nombreux algorithmes toujours dans le domaine du multimédia. Une compression efficace des images en est un exemple. La compression des composantes plates, "cartoon", et de texture séparément peut donner de meilleurs résultats [17]. Cette décomposition peut être utilisée pour le débruitage des images [7]. En effet, le bruit peut être considéré comme une texture fine et indépendante de la composante structurelle de l'image.

Les méthodes de déconvolution sont largement utilisées en traitement d'images où la défocalisation est un phénomène courant: de la microscopie à l'astronomie. Même si aujourd'hui la régularisation est la tendance principale dans ces approches, les méthodes sans régularisation sont également capables de produire des résultats comparables à ceux de l'état de l'art [20]. Pour ces méthodes sans régularisation, le problème d'arrêt est une question clé. La méthode que nous avons proposée offre une solution logique et efficace à ce problème.

Les perspectives de ces travaux sont nombreuses. Premièrement et de façon évidente les méthodes de déconvolution proposées pourraient être appliquées en guise de pré-traitement aux vidéos acquises avec des caméras portables pour une meilleure qualité de visualisation mais aussi pour en faciliter le traitement.

La séparation en composantes plates et de texture nous semble prometteuse dans le contexte de recalage des images en mouvement. En effet, les méthodes communes d'estimation du mouvement dans le cas des images bruitées, que cela soit les méthodes dite de "flux optique" ou de mise en correspondance amènent à des résultats erronés. Les travaux récents montrent l'intérêt d'introduction de contraintes basées image. Dans ce contexte, disposer de la composante structurelle nous semble une voie prometteuse pour une meilleure qualité d'estimation.

En ce qui concerne la séparation des objets d'avant-plan animés par

DOI:10.15774/PPKE.ITK.2011.005

mouvement propre, il ne s'agit que de la première étape dans le problème complexe d'identification-reconnaissance des objets dans des contenus vidéo. Ce problème, actuellement intensivement exploré par la communauté de recherche en multimédia et en vision par ordinateur, est l'un des plus importants et notre méthode doit être inscrite dans le cadre de ces recherches.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 1

Introduction

The first applications of digital imaging dates back to the early 1920s, when coded images were transfered through a submarine cable between London and New York. In the 1960s the improvement of computing technology and the beginning of the space race motivated a new wave of research in digital image processing. The first space photographs of the Moon were enhanced with digital image processing techniques and a decade later medical applications gave a new motivation to the researchers in the field. In the last 30 years image processing has become a mature engineering discipline and it has become an indispensable tool for many fields like medical visualization, law enforcement, human computer interaction, industrial inspection and security or medical surveillance.

The evolution of technology in the last decade opens up new possibilities, and the new possibilities set up new challenges. In the early days of digital image processing there were only digital images to process in a relatively low number. Around the turn of the millennium videos appeared and in parallel the constantly growing size of the image databases exceeded the manually manageable limit. New methods were required to handle the new challenges: content based image retrieval, video coding, event detection in videos have become part of digital image processing. Nowadays everyone can easily access digital cameras and make digital video recordings, hence the amount of video data is rapidly increasing. At the same time the type of video content has become more challenging, since generally neither the "cameraman" nor the device is professional. Blurry, noisy recordings with practically random camera motion need to be analyzed. Obviously to detect events in these kinds of recordings the whole process from low- to high-level has to be adapted to the task.

This work is concerned with low- and mid-level image processing problems, that need to be solved to handle these new kinds of videos efficiently. The first two parts of the dissertation address basic image enhancement problems such as optimizing deconvolution for image deblurring, and extraction of the geometrical structure of the image by decomposing it into texture and geometrical components, while in the third part, higher level video understanding will be examined,

DOI:10.15774/PPKE.ITK.2011.005

where the task is the detection of moving objects and their separation from a cluttered background in videos recorded with a moving camera.

Image restoration is practically as old as image processing itself, constantly waiting for newer and better solutions. Deconvolution of blurred images, like the ones taken with strongly moving wearable cameras, gives a new motivation to solve an old challenge. Beside motion, there could be many other reasons of image blur like defocusing, atmospheric perturbations, optical aberrations. For these reasons, which are common in aerial, satellite or medical imaging, the acquired images are corrupted and restoration is needed. The distortion of the image is generally modeled as convolution: the original unknown image is convolved with a Point Spread Function (PSF) that describes the distortion. The goal is obvious: restore the original image as well as possible based on the blurry measurement and, in some cases, the PSF. The problem is ill-posed, since there is more than one image that would seem as a good solution. Hence it is a common drawback of non-regularized iterative deconvolution methods that after some iterations they start to amplify noise (see Fig. 1.1). Our goal was to automatically find an optimal stopping condition for these algorithms where the reconstructed image is as close to the original (unknown) image as possible.

(a) Blurred Image (b) Reconstructed image after 14 iterations (c) Reconstructed image after 60 iterations

Figure 1.1: An example how non-regularized deconvolution methods amplify noise if not stopped at the optimal iteration.

The decomposition of an image into geometrical (cartoon) and noise like (texture) components is a fundamental task for both videos and still images. It can help image compression, denoising, image feature selection or it can be a pre-

DOI:10.15774/PPKE.ITK.2011.005

processing step for video event detection: the same way as shadow, reflection or smoke/fog removal, the elimination of texture from the video frames aids the higher level understanding of the video. Theoretically the two parts are independent of each other: the cartoon image contains only geometrical information while the texture image, as complementary of the cartoon, is free of geometrical information (see an example on Fig. 1.2).

(a) Artificial original image (b) Cartoon component (c) Texture component

Figure 1.2: An example of cartoon/texture decomposition.

Separating foreground objects from the background is a fundamental module for many video applications, as it is commonly used to bootstrap higher-level analysis algorithms, such as object-of-interest detection, tracking, or content based video indexing, which could be applied for security or medical surveillance. The task is challenging for still camera recordings, but if wearable cameras are used, then strong motion and parallax, low quality of signal (reduced by motion blur) makes the problem even more complex. Generally low level algorithms, like the ones presented in Part I and II of the dissertation, can be used as preprocessing to achieve better results.

DOI:10.15774/PPKE.ITK.2011.005

Part I

Optimal Stopping Condition for Iterative Image Deconvolution

DOI:10.15774/PPKE.ITK.2011.005

Deconvolution techniques are widely used for image enhancement from microscopy to astronomy. The most effective algorithms are usually based on some iteration techniques. The determination of the optimal stopping condition is a common problem for non-regularized methods. In this part of the dissertation an automatic procedure is presented for estimating the stopping condition based on a special independence measure that checks an orthogonality criterion of the estimated signal and its gradient at a given iteration. An effective lower bound estimate is provided to the conventional *ad-hoc* non-regularized methods, proving its superiority against the others for a wide range of noise levels at different noise models.

This part of the dissertation begins with the general presentation of the image blurring problem and deconvolution methods presented in the literature. Afterwards the more specific iteration stopping problem of the non-regularized iterative deconvolution algorithms will be introduced. In Chapter 3, the introduction of the proposed theoretical solution to this problem is described, which is followed by the obtained results (Chapter 4) and the conclusions (Chapter 5).

DOI:10.15774/PPKE.ITK.2011.005

Chapter 2

Problem Statement

In almost all image acquisition processes blurring is a common issue. Due to various reasons (like motion, defocusing, atmospheric perturbations, or optical aberrations), the acquired images are distorted, and without restoration they are often useless. The distortion is generally modeled as a convolution: the original unknown image is convolved with a Point Spread Function (***PSF***) that describes the distortion that a theoretical point source of light undergoes through the image acquisition process.

$$Y = H * U + N, \tag{2.1}$$

where Y is the measured blurry image, U is the unknown original image, H is the PSF and N is additive noise with zero mean. Y, U, N are (n, m)-sized 2D images, H is a (k, l)-sized kernel ($k \leq n, l \leq m$) with some boundary constraints, and $*$ denotes 2D convolution. On a pixel-wise basis (2.1) can be rewritten as:

$$Y(x, y) = \sum_{x'=1}^{k} \sum_{y'=1}^{l} U(x', y') H(x - x', y - y') + N(x, y), \tag{2.2}$$

where (x, y) and(x', y') are pixel positions.

2.1 Overview of Deconvolution Methods

2.1.1 Linear Methods

The convolution of an image with a PSF in the spatial domain is equivalent to the multiplication of the Fourier transform of the image with the Fourier transform of the PSF, also known as the Optical Transfer Function (***OTF***). Therefore, a naive form of image restoration is to divide the Fourier transform of the image by the OTF. This procedure is known as inverse filtering [21]:

$$X(x, y) = F^{-1} \frac{\hat{Y}(\omega_x, \omega_y)}{\hat{H}(\omega_x, \omega_y)} \tag{2.3}$$

DOI:10.15774/PPKE.ITK.2011.005

where X is the deconvolved image with (x,y) position variables, $\hat{Y}$ and $\hat{H}$ are the Fourier transforms of Y acquired image and H PSF respectively, (ω_x, ω_y) are the counterparts of (x, y) in the frequency domain, and F^{-1} denotes the inverse Fourier transform.

Inverse filtering takes only blurring into account and does not handle stochastic distortion, which results in the amplification of high-frequency noise [22].

A common way to restore blurring in the presence of noise is to use regularization in the restoration procedure. The Tikhonov [23] and Wiener [24, 25] filters are both regularized linear deconvolution filters. The Tikhonov filter is a linear restoration filter that minimizes the Tikhonov functional, while for signal independent additive Gaussian noise, the Wiener filter is the Mean Square Error-optimal stationary linear filter for deconvolution. They regularize their results by restoring the frequencies that are dominated by the object while suppressing those frequencies that are dominated by noise. This way the problems that arise when using the inverse filter are avoided.

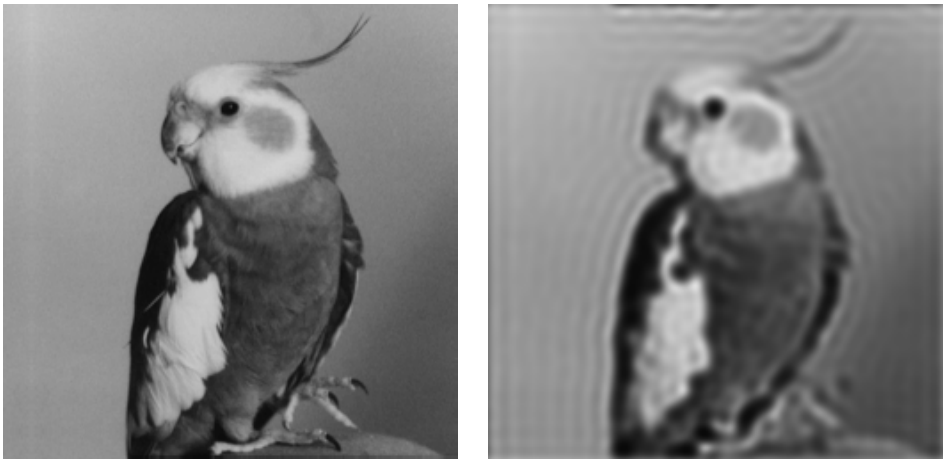
The above mentioned Mean Square Error (***MSE***) is a commonly used error measure for deconvolution algorithms. If Q and P are two images of the same size, then $MSE(Q, P)$ can be defined as follows:

$$MSE(Q,P) = \frac{1}{n \cdot m} \sum_{x=1}^{n} \sum_{y=1}^{m} |Q(x,y) - P(x,y)|^2 , \tag{2.4}$$

where $|.|$ is the Euclidean norm.

The problem with linear space-invariant filters [21] is that they cannot restrict the intensities in the restored image to positive values and sometimes they estimate negative intensity in the deconvolved image. Unlike superresolution methods [21, 26] they can only restore frequencies inside the bandwidth of the OTF. In addition, these methods are very sensitive to errors in the estimation of the PSF used for the restoration. The utilization of an imprecise PSF may cause a ringing artifact in the deconvolved image (see Fig. 2.1.).

DOI:10.15774/PPKE.ITK.2011.005



(a) Original image (b) Reconstructed image with ringing artifact

Figure 2.1: The illustration of the ringing artifact.

2.1.2 Nonlinear Methods

To handle the difficulties encountered using linear methods, nonlinear methods were developed with additional constraints (e.g., nonnegativity, finite support, smoothness, and regularization) about the target image. The cost of the better quality is the increased computational complexity.

The Janson-Van Cittert (***JVC***) [27] algorithm was the first iterative method for constrained deconvolution to prevent negative intensities or very bright intensities. It amplifies high-frequency noise, hence it requires a smoothing step at each iteration. Unfortunately, the smoothing operation does not work well for low Signal to Noise Ratio (***SNR***) images [28]. The JVC algorithm was introduced to light microscopy by Agard in 1989 [29]. The JVC filter in spatial domain was also used in [30] with regularization for super-resolution mosaicking from video data.

The error between the observed and the estimated image can be negative for the JVC method. In the nonlinear least-squares (***NLS***) approach the sum of the squared error is constrained by nonnegativity where either negative values are set to zero or the estimated image is constrained to be positive [31].

The iterative constrained Tikhonov-Miller (***ICTM***) algorithm [32,33] and the Carrington algorithm [26, 31, 34] are both non-linear algorithms that iteratively

DOI:10.15774/PPKE.ITK.2011.005

minimize the Tikhonov functional with conjugate gradient descent algorithm. Both algorithms are based on an additive Gaussian noise model, but differ in the way they incorporate the non-negativity constraint. In 1997 Verveer [35, 36] proposed an optimization to the conjugate gradient descent algorithm for minimizing the Tikhonov functional and a quadratic transformation to incorporate the non-negativity constraint.

Tikhonov filtering, NLS, ICTM, and the Carrington algorithms are based on the assumption that the noise can be modeled as additive Gaussian noise, but they do not have a direct noise-reduction strategy with an *a priori* noise model. Statistical processing with necessary physical constraints would eliminate both the out-of-focus light and random noise and, thus improve the deconvolution performance [37].

2.1.3 Statistical Methods

In the presence of strong noise in the acquired image, statistical methods are very effective [22]. These methods have a more sophisticated noise strategy than the simple regularization. However, they are more complex and computationally more expensive than the linear and nonlinear methods.

Richardson and Lucy [2, 3] proposed a non-linear, iterative image restoration algorithm. It is a non-regularized Maximum Likelihood Estimator (***MLE***) for the intensity of a Poisson process and it produces a positive-constrained restoration result. Later, Holmes [38–41] introduced a deconvolution algorithm to fluorescence microscopy also based on the Expectation-Minimization Algorithm (***EM***). This algorithm has been developed as a reconstruction algorithm for computer tomography [42]. The algorithm iteratively finds the MLE when the image is distorted by Poisson noise. The drawbacks of EM algorithm are its slow rate of convergence and its computational complexity.

The EM algorithm is also very sensitive to the noise present in the acquired image [43]. To reduce this sensitivity, several regularization methods were proposed [44–46]. Recent work by Zou [20] has proved that excellent restoration results can be achieved using non-regularized RL algorithm performing some simple post processing.

DOI:10.15774/PPKE.ITK.2011.005

2.1.4 Blind Deconvolution Methods

All the previous algorithms depend on an estimate of the PSF, which is a challenging task for real images. Noise is always present in an experimentally measured PSF, whereas a theoretical PSF cannot completely determine all aberrations present in microscope optics [37].

Based on the Richardson-Lucy (***RL***) algorithm, several authors [22, 41, 47–49] have proposed Blind Deconvolution algorithms (***BD***) that restore both the original object and the PSF from the measured image, therefore they do not require *a priori* knowledge about the PSF.

To reduce the noise sensitivity of the RL algorithm, the high-frequency parts are suppressed by convolving the acquired image with a Gaussian function. This convolution operation causes smoothing in the RL algorithm and is further compensated by convolving the PSF with the same Gaussian [43]. There are many variations of BD in the literature (for instance simulated annealing, error metric minimization, ML method, EM algorithm, among others), all of them estimate simultaneously and iteratively the original image and the PSF.

2.1.5 Description of the Method Used in the Experiments

This part of the dissertation focuses on a common issue of non-regularized iterative methods, namely the stopping condition. Although nowadays regularization is the main trend, non-regularized methods also produce results comparable to the state-of-the-art [20]. A new error metric is described based on the independence of the estimated signal and the estimation noise and we show that the ideal (but unknown) criterion can be well estimated by this theoretically new calculus.

In the experiments an iterative, standard, non-blind deconvolution algorithm, an accelerated Richardson-Lucy method was used, described in [50, 51].

As it is stated in [50] the RL algorithm is an iterative technique used for the restoration of astronomical imagery in the presence of Poisson noise. It attempts to maximize the likelihood of the restored image by using the EM algorithm. It requires a good estimate of the process by which the image is degraded for accurate restoration. The degradation can be caused in many ways such as subject movement, out-of-focus lenses, or atmospheric turbulence, and is described by

DOI:10.15774/PPKE.ITK.2011.005

the PSF of the system. The image is assumed to come from a Poisson process. There may also be other forms of noise involved in the image acquisition process (electronic or quantization noise). The image degradation can be described by (2.1).

The standard equation of the RL method is as follows:

$$X_{k+1} = X_k \left(H * \frac{Y}{\tilde{Y}_k} \right) \tag{2.5}$$

where X_k is the estimated reconstructed image after k iteration H is the known point spread function and Y is the acquired, distorted image and $\tilde{Y}_k$ is the re-blurred estimated image after k iterations:

$$\tilde{Y}_k = H * X_k \tag{2.6}$$

As we an see in (2.5) the restoration is based on the reblurred estimated image, therefore only the $\tilde{Y}_k$ converges to Y as k goes to infinity. In the next section it will be shown, that it does not guarantee the convergence of $\tilde{X}_k$ to the original image.

2.2 Necessity of the Stopping Condition for Iterative Methods

Since we do not know the original image U, only the blurry measured image Y can be used to guide us toward U. If $X(t)$ is the output of the deconvolution process after t iterations (starting with $X(t=0) = Y$), then a cost function of the method is usually based on minimizing the MSE:

$$MSE(Y, H * X(t)) = \frac{1}{n \cdot m} \left| Y - H * X(t) \right|^2, \tag{2.7}$$

where $|.|$ is the Euclidean norm.

The above MSE measures similarity between two images. In the ideal case the goal is to minimize $MSE(U, X(t))$ (or $|U - X(t)|$) by stopping the iterations at the minimum. However, we can only access the smoothed and thus vague

DOI:10.15774/PPKE.ITK.2011.005

$|Y - H * X(t)|$. Let $X(t)$ be an iterated estimation, while another one is $X'(t) = X(t) + N(t)$, where they differ by an additive noise $N(t)$ and residual error with zero mean. In this case $H * N(t) \approx 0$ can be considered, and $H * X'(t) = H * X(t) + H * N(t) \approx H * X(t)$. Since the iterations of $X(t)$ are controlled by $H * X(t)$, this allows possible cases for $t_n \neq t_m$ where $|X'(t_n) - U| >> |X(t_m) - U|$ is true, while $|H * X'(t_n) - Y| \leq |H * X(t_m) - Y|$ (see Fig. 2.2), and this is why the problem is ill-posed. As stated in [37, 43, 52], this problem affects the quality of solution of the iterative algorithms highly.

2.3 Techniques Related to the Iteration Stopping Problem

Most deconvolution algorithms in current use incorporate regularization to stop the corruption of the estimate described in the previous section [44–46, 53, 54]. Through a regularization parameter these algorithms balance the need to fit the restoration result to the acquired image and to an *a priori* model at the same time [55]. A large value of the regularization parameter results in a stronger influence of the regularization on the restoration result, whereas a low value will make it more sensitive to the noise. Therefore, it has a large influence on the produced result [56].

The current work will show that a theoretically optimal result can be achieved without any regularization, based on a noise independence criterion only.

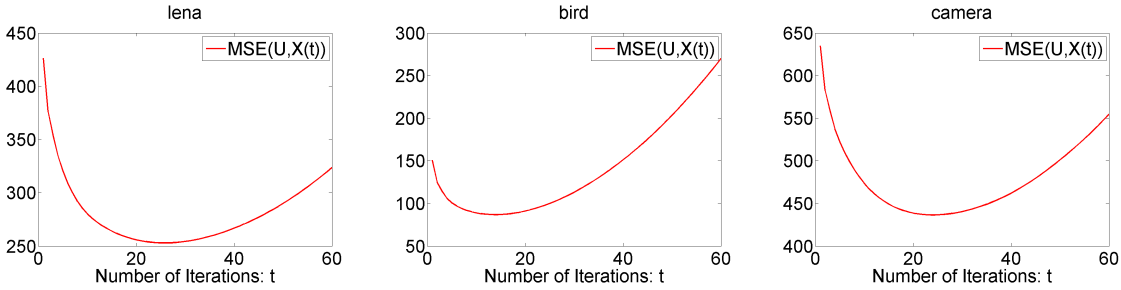
Without regularization the best way to stop the corruption of the reconstructed image is to estimate the number of iterations needed to reach the best image quality and stop the process there. A straightforward idea is to stop the process after a constant number of iterations. Based on our experiments, this constant is around 7-10 iterations for a Lucy-Richardson based method [2, 3] like the one we used [50, 51]. For other methods this constant is different (see [52]). Obviously the optimal stopping point depends on many factors (for instance the image itself, the PSF, the noise level, and so on), hence using a constant number of iterations for all the different cases will many times result in poor image quality.

Another way to prevent the corruption of the estimate is to stop the iteration after the change of the image between two consecutive estimations becomes lower

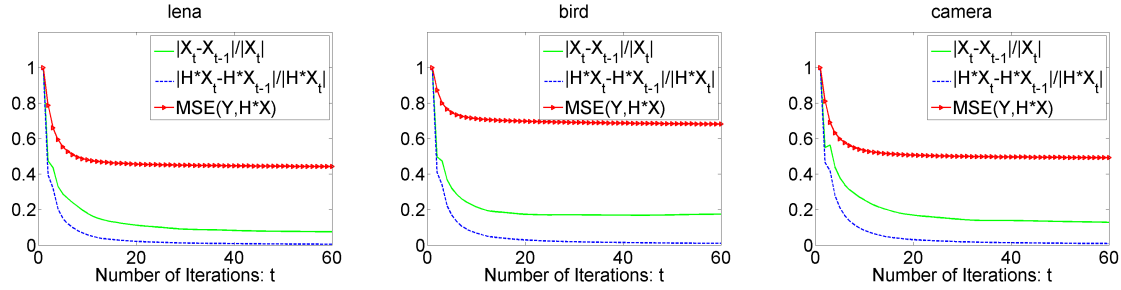
DOI:10.15774/PPKE.ITK.2011.005

(a) Original Images

(b) Blurred Images



(c) MSE Functions



(d) Other Measurable Functions (relative values)

Figure 2.2: Tree example images show that the measurable function $MSE(Y, H*X(t))$ and other investigated methods do not follow the unmeasurable function $MSE(U, X(t))$.

DOI:10.15774/PPKE.ITK.2011.005

than a certain threshold [54]. In the following we will call this Differential Based Stopping Condition (***DBSC***):

$$DBSC: \qquad \frac{|X(t) - X(t-1)|}{|X(t)|} < th \tag{2.8}$$

where th is a heuristic choice for threshold, usually between 10^{-3} and 10^{-6}. We have also tested a modified version of the above condition (in the following: ***MDBSC***), where the re-blurred estimated images, $H * X(t)$ were considered instead of $X(t)$:

$$MDBSC: \qquad \frac{|H * X(t) - H * X(t-1)|}{|H * X(t)|} < th \tag{2.9}$$

Other similar stopping criteria are summarized in [52]. The problem with all these methods is that the number of iterations needed to reach the optimal restored image depends on many other things: the picture itself, the PSF and the additional noise. See Fig. 2.2(c).

DOI:10.15774/PPKE.ITK.2011.005

Chapter 3

Orthogonality Based Stopping Condition

When we are searching for the optimal stopping condition of an iterative image deconvolution method, we are looking for the iteration t for which the quality of the reconstructed image is the best. This condition can be expressed using the $MSE(U, X(t))$ function as quality measure as follows:

$$\min_t \left(MSE\left(U, X(t)\right)\right) \tag{3.1}$$

Since U is unknown it is impossible to calculate the above function directly, therefore we search for a function that finds the minimum close to (3.1) but uses only known images.

3.1 Angle Deviation Error Measure

In recent years a new estimation error has been introduced for focus measurement in blind deconvolution problems, see [1]. This error definition, called Angle Deviation Error (***ADE***), is based on the orthogonality principle [57], considering the independence of noise and the estimated signal, using the scalar product:

$$ADE(Q, P) = \left| \arcsin \left(\frac{\langle Q, P \rangle}{|Q| \cdot |P|} \right) \right|, \tag{3.2}$$

where $\langle . \rangle$ is the inner product, Q and P are $n \times m$-sized vectors on $\mathbb{R}$. We will also show that conventional measures, like MSE, cannot help us to find optimal stopping criteria; while ADE has an optimum, close to the minimum of (3.1).

The logic behind the construction of the ADE is to get back the complementary angle of the angle between the two input vectors. The inner product of two vectors Q and P in Euclidean geometry can be expressed with the length of the vectors and the θ angle between them as follows:

$$\langle Q, P \rangle = |Q| \cdot |P| \cdot \cos\theta, \tag{3.3}$$

DOI:10.15774/PPKE.ITK.2011.005

Using the inner product as measure of independence would make the result depend on the magnitude of the component vectors, which is undesirable. The angle on the other hand would make a clean independence measure. Expressing θ from the above equation we get the following:

$$\theta = \arccos\left(\frac{\langle Q, P\rangle}{|Q| \cdot |P|}\right), \tag{3.4}$$

This measure is almost the same as the ADE except for two parts: we calculate arcsin(.) instead of arccos(.) – hence we will receive $\pi/2 - \theta$ instead of θ – to make the ADE response for perfect independence to be 0, and we apply absolute function on the result since negative values express dependency as well as positive values and our goal is to find independence.

The received measure is similar, but not the same as standard correlation, where zero-mean vectors are used to calculate the scalar product and the normalization is done with the standard deviation of the vectors:

$$corr(Q, P) = \frac{cov(Q, P)}{\sigma_Q \cdot \sigma_P} = \frac{\sum_{i=1}^{n}(Q_i - \mu_Q)(P_i - \mu_P)}{n \cdot \sigma_Q \cdot \sigma_P} \tag{3.5}$$

where $cov(.)$ is the covariance over the elements of vectors, σ_Q, μ_Q and σ_P, μ_P are the standard deviation and the expected values of the elements of Q and P respectively, and n is the size of the vectors.

Comparing the two measures, we can see that they are very similar: if both Q and P had zero mean, the two measures would actually give the same result. In this dissertation we always use the ADE with two input vectors out of which only one has inherent zero-mean while the other does not. For these ADE is better suited and also faster to compute then the normalized correlation, since we do not have to calculate the vectors mean and standard deviation.

3.1.1 Use of ADE Measure for Focus Estimation

In [1] the authors use localized blind deconvolution on small blocks of an image to estimate focus area (see the examples in Fig. 3.1). This method could be useful for content based indexing of images. However, the ill-posed iteration process of the deconvolution tends to be noisy with higher number of iterations. The focus

DOI:10.15774/PPKE.ITK.2011.005

depth classification was based on the the distortion of a spot, the more in the focus a spot is, the higher the distortion is at an early iteration. An error measure was needed which consistently gives different values for differently focused areas, and which is not much affected by the process's noisy nature. Their experiments have shown that MSE is not suitable for the task since it is sensitive to the noise coming from the ill-posedness of the iteration process which caused fluctuations in the classification. Thus, a more stable error measure was introduced. This measure theoretically converges to zero and instead of simple block differences gives the angle deviation error (ADE) of the measurement and estimation residual error. The main reason ADE has proven to be better suited for the focus depth classifications is that, while MSE is a simple difference measure, which can greatly vary and cannot provide a consistent scale, ADE gives the normalized angle of the reconstruction error.

Figure 3.1: Examples for focus extraction on various images [1]. The top row shows the input images while the bottom row shows respective focus maps.

3.2 The ADE Measure as Stopping Criterion for Deconvolution Algorithms

In our case the problem with MSE is that it measures similarity between two images. Since U is unknown, we cannot calculate $MSE(U, X(t))$, the only remaining logical possibility is to use Y and $H * X(t)$. These two values are the base of the goal function of the iterative deblurring algorithms, which leads us back to the original ill-posed problem, meaning that the $MSE(Y, H * X(t))$ will decrease monotonously, although after a while the quality of the reconstructed image will start to decrease as a result of high frequency noise appearing on the

DOI:10.15774/PPKE.ITK.2011.005

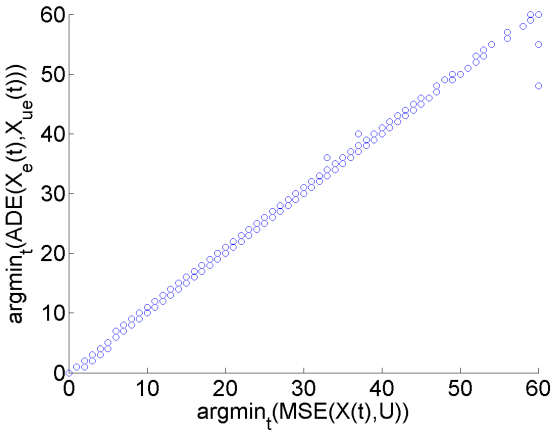


Figure 3.2: The relationship between the minimum of $MSE(U, X(t))$ and $ADE(X_e(t), X_{ue}(t))$ for various pictures with different SNR and blur radii.

image $X(t)$ (see Fig. 2.2(c) for illustration).

3.3 The ADE Function as Stopping Criterion

In general we cannot measure the noise from the measurement Y, so noise-model based approaches are limited for the general use. However, when estimating the $X(t)$ image, there is a point when further iterations do not enhance the image anymore and the difference between two consecutive estimated images, $X_e(t) = X(t) - X(t-1)$ contains minimal information about the unknown residual error of the estimated image vs. the original, $X_{ue}(t) = X(t) - U$. We can assume at this point the independence of $X_e(t)$ and $X_{ue}(t)$ to be maximal.

Once the independence has been reached, the process must be stopped, since any steps after this point may add false information – which is not part of U – to the reconstructed image. In other words we have to stop the iterative process when $ADE(X_e(t), X_{ue}(t))$ function reaches its minimum.

This theory has been confirmed by checking the statistical dependency between $ADE(X_e(t), X_{ue}(t))$ and $MSE(U, X(t))$: the $\arg\min_t ADE(X_e(t), X_{ue}(t))$ correlates well with the $\arg\min_t, MSE(U, X(t))$, the correlation coefficient is 0.9986 for our image database, see Fig. 3.2 (the details of the database can be found in the Results chapter).

DOI:10.15774/PPKE.ITK.2011.005

$ADE(X_e(t), X_{ue}(t))$ still refers to the unknown image U. $Y - H * X(t)$ cannot be used instead of the unknown $X_{ue}(t)$, since the deviation error is blurred by the function H in $H * X(t)$. The clearest way of capturing the independence of the signal and the noise is using the difference between two consecutive estimated images $X_e(t)$ and the unblurred estimation $X(t)$:

$$ADE(X_e(t), X(t)) = \left| \arcsin \left(\frac{\langle X_e(t), X(t) \rangle}{|X_e(t)| \cdot |X(t)|} \right) \right| \tag{3.6}$$

The above expression contains only measurable images and provides a reasonable solution to the stopping problem. At the minimum of $ADE(X_e(t), X(t))$ the change between two consecutive iterations $X_e(t)$ has the highest possible independence of the actual reconstructed image, hence we can assume that at this point $X_e(t)$ contains mostly independent noise and not structural information about the image; further iterations will not enhance the image quality, but may add more noise. Although, the correlation is weak (0.6726) related to that of between $ADE(X_e(t), X_{ue}(t))$ and $MSE(U, X(t))$, it is clearly visible (see Fig. 3.3). In the following we will examine how close this approximation brings us to the optimal stopping point.

3.3.1 Theoretical Explanation

In an ideal case, when $X_e(t_n)$ contains only random noise and $\langle X_e(t_n), X(t_n) - U \rangle = 0$ for a given t_n (which means that the iterated change is independent of the structural differences of the restored image), then, using the distributive property of the scalar product gives us the following:

$$\langle X_e(t_n), X(t_n) - U \rangle = \langle X_e(t_n), X(t_n) \rangle - \langle X_e(t_n), U \rangle. \tag{3.7}$$

Since U may contain high frequency components correlating with X_e, the possible zeros of components in (3.7), $\langle X_e(t_n), X(t_n) - U \rangle = 0$ and $\langle X_e(t_m), X(t_m) \rangle = 0$, are not necessarily coincident, since $\langle X_e(t), U \rangle \neq 0$ biases (3.7): $t_m \neq t_n$. We may say that, even for the most ideal case, the ill-posed property of the problem results in the biasing of the possible minimum, $t_n \neq t_m$.

DOI:10.15774/PPKE.ITK.2011.005

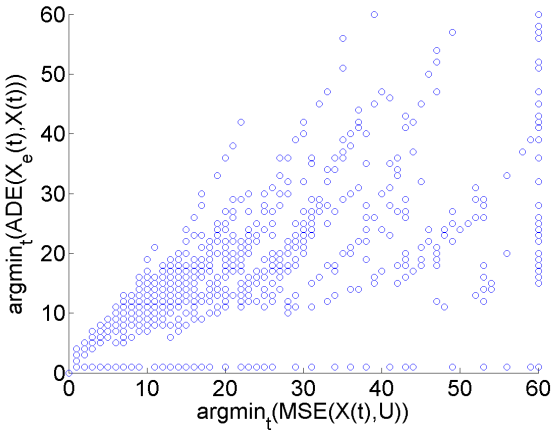


Figure 3.3: The relationship between the minimum of $MSE(U, X(t))$ and $ADE(X_e(t), X(t))$ for various pictures with different SNR and blur radii. In some of the cases the ADE function is monotonically increasing or the MSE is monotonically decreasing through the 60 iterations, which causes the horizontal line of dots at 0 and the vertical line of dots at 60.

3.3.2 Quality of the Proposed Stopping Condition

As has been mentioned earlier, the correlation between the $\arg\min_t ADE(X_e(t), X(t))$ and the $\arg\min_t MSE(U, X(t))$ is weaker than between the $\arg\min_t ADE(X_e(t), X_{ue}(t))$ and the $\arg\min_t MSE(U, X(t))$. This correlation only shows how close the proposed stopping point is to the ideal one regarding the iteration number. However, what really matters is not the difference in the number of executed iterations, but the difference between quality of the estimated image.

A generally accepted way to quantify the quality of the estimation is to compare the MSE value between the original image (only available under test circumstances) and the reconstructed one. Our main priority is to provide an estimation $X(\alpha)$ that is as close to the theoretically best iteration $X(\beta)$ as possible, where $\beta = \arg\min_t(MSE(U, X(t)))$ and $\alpha = \arg\min_t(ADE(X_e(t), X(t)))$.

The shape of the $MSE(U, X(t))$ as the function of the iteration number t changes from image to image, and the blur radius or the SNR of Y may also affect it, therefore the iteration distance between β and α is not the best measure for us. It is possible that for m_1 measurement (a given image, blur radius and

DOI:10.15774/PPKE.ITK.2011.005

SNR) the difference between β and α is bigger than for an other measurement m_2 and in the same time the difference between $MSE(U, X(\beta))$ and $MSE(U, X(\alpha))$ is smaller for m_1 then for m_2.

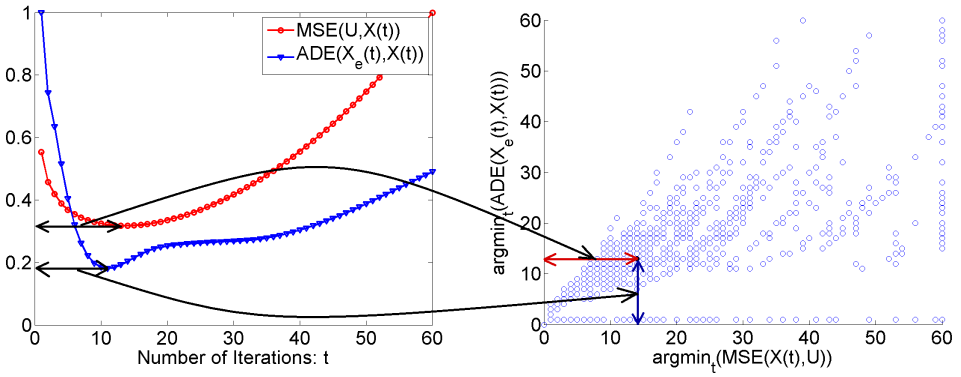
Fig. 3.3 shows the relation between β and α for different measurements. Each point corresponds to a measurement and the coordinates of the point are (α, β). Calculating the average $E\{MSE(U, X(t))\}$ function for a given $\alpha = \arg\min_t(ADE(X_e(t), X(t)))$, the minimum of the resulting average curve is considered as the best common stopping point γ:

$$\gamma = \arg\min_t(E\{MSE(U, X(t))\}). \tag{3.8}$$

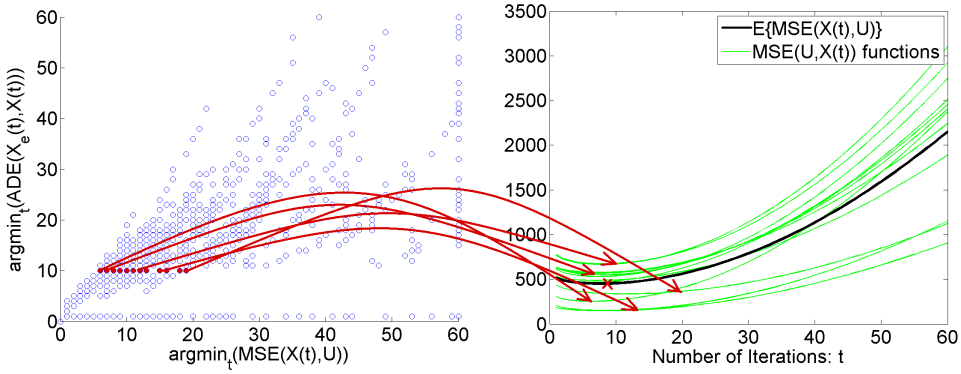
The calculated γ locations can be seen versus the proposed iteration stopping points, $\alpha = \arg\min_t(ADE(X_e(t), X(t)))$ in Fig. 3.4(c). We may see that for the measurements, which were stopped at the same iteration by the proposed method - they have the same α value - the expected value of β is close to α.

Fig. 3.4 illustrates the calculation of a γ point. The right side of Fig. 3.4(c) shows that the correlation between α and γ is high, which indicates that the quality of the image reconstructed with the proposed method is closer to the theoretically optimal value, than the different iteration counts (α and β) would suggest.

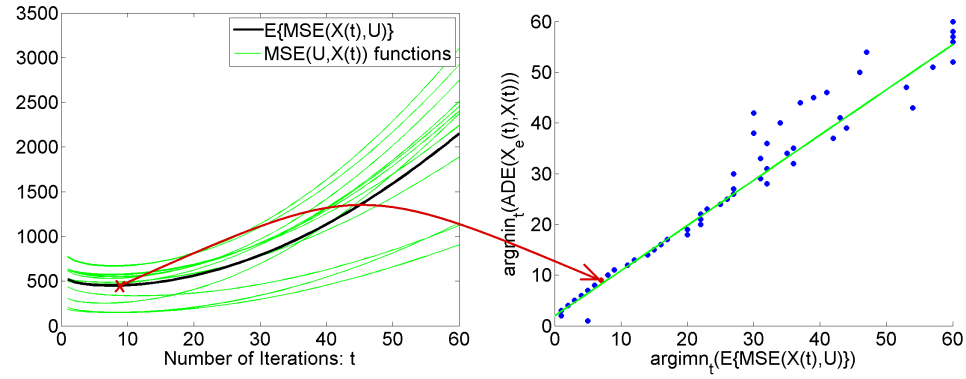
DOI:10.15774/PPKE.ITK.2011.005



(a) The relationship between the minimum of $MSE(U, X(t))$ and $ADE(X_e(t), X(t))$ for measurements of various pictures with different SNR and blur radii can be seen on the right. The left side shows an example of one measurement, its $MSE(U, X(t))$ and $ADE(X_e(t), X(t))$ functions and how it is added to the figure on the right.



(b) An illustration of the calculation of a best common stopping point (γ) for measurements in the same row.



(c) The calculated γ locations versus the proposed stopping points, $\alpha = \arg\min_t(ADE(X_e(t), X(t)))$

Figure 3.4: An alternative quality measure for the proposed method based on MSE values.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 4

Results

To test the proposed method and to compare it to other algorithms, we used a database of 25 images containing landscapes, images of buildings, animals, textures as well as black and white drawings. The PSF is a Gaussian kernel defined by different blur radii between 1 and 5 pixels. We tested a wide range of noise levels and different models: Poisson noise or white Gaussian noise with SNR=20, 25, 30, 35, 40dB was added to the blurred images.

4.1 Comparative Results

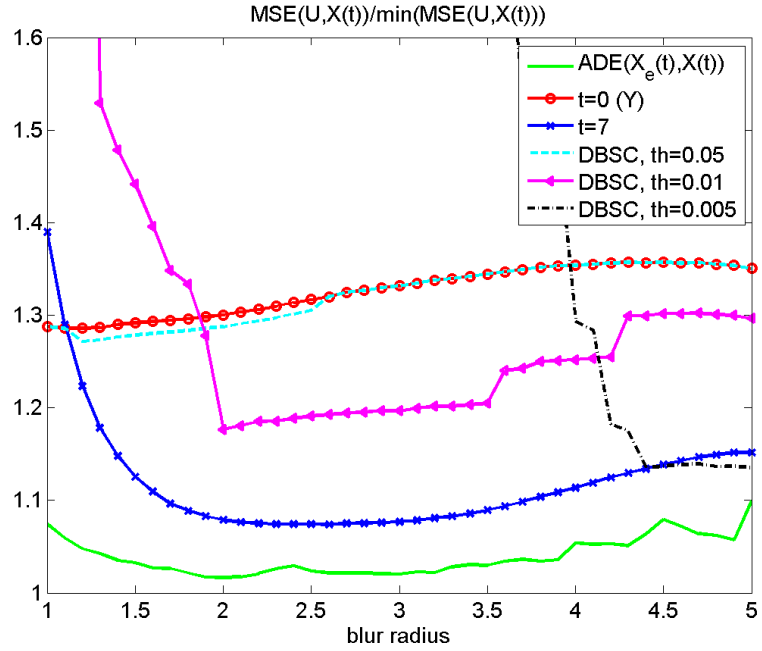
To compare the proposed method to other existing stopping conditions, we calculated the ratio between the MSE value at the real minimum location (see Fig. 2.2(c)) and at the point where the stopping condition would stop the iteration.

We compared the proposed method to fixed iteration count (the best results were obtained when this constant was 7), the DBSC (Eq. (2.8)) and as a baseline we also calculated the above mentioned ratio for the blurred image, Y. The experiments were taken using all the 25 images with different blur radii and noise levels. The results can be seen on Figs. 4.1.

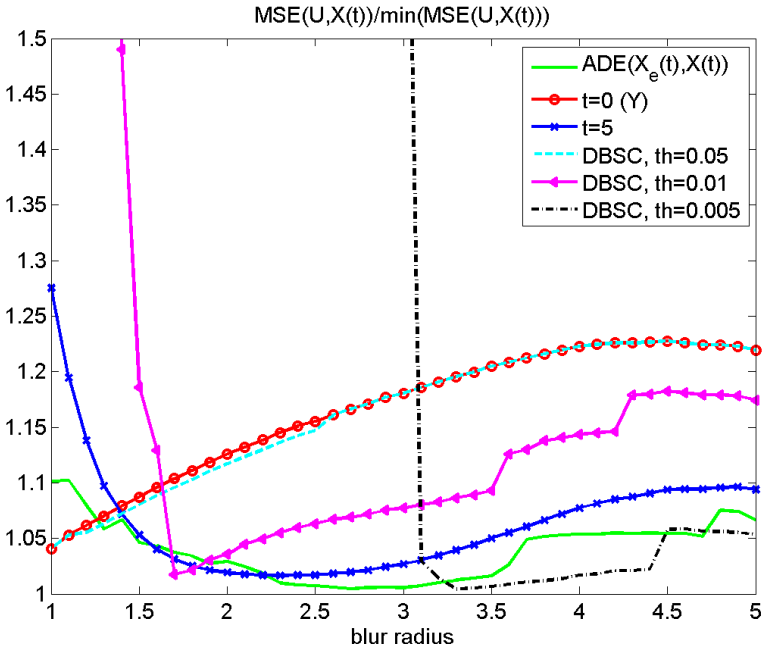
Our tests prove that the commonly used X_e-based (DBSC) stopping condition is outperformed by the one using the blurred comparison (MDBSC). Both of them are outperformed by the proposed ADE based function.

We have tested the stability of the proposed and compared methods at different noise levels between 20 and 40 dB (see Fig. 4.2(a)) and against inaccurate estimation of the radius of H Gaussian deconvolution kernel (see Fig. 4.2(b)). The results show that our $ADE(X_e(t), X(t))$ stopping criterion gives the best SNR estimation of U along with a well balanced run-time effort at each noise level and it is the most robust if the radius of the deconvolution kernel deviates from the size of the blurring kernel by less than 10%, which is considered a reasonable assumption. Fig. 4.3 shows estimation of the original image U at different t points of the iteration process.

DOI:10.15774/PPKE.ITK.2011.005



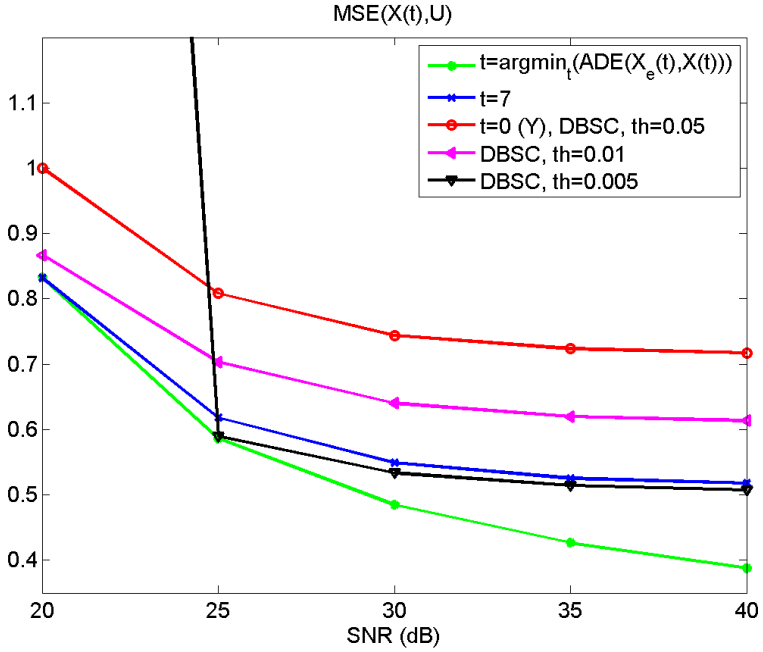
(a) Results with Gaussian Noise



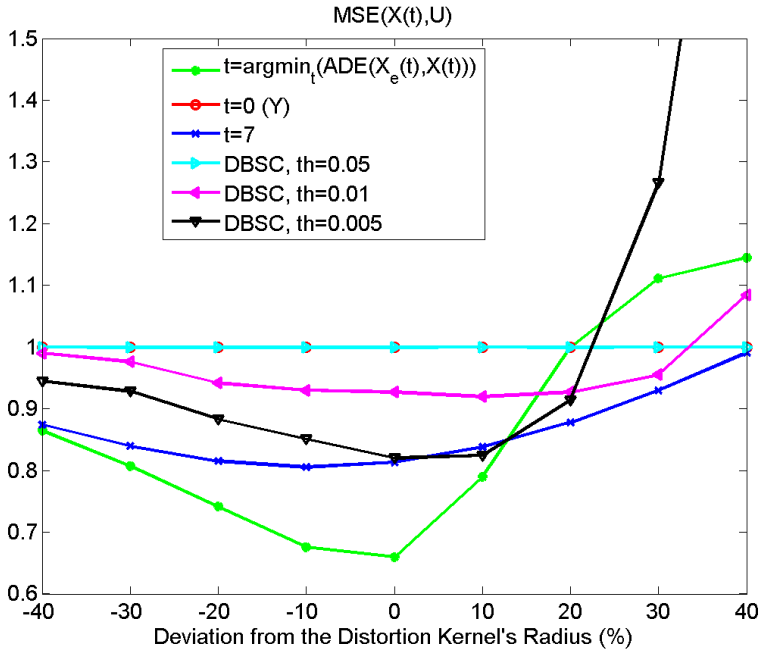
(b) Results with Poisson Noise

Figure 4.1: The figure shows the relative MSE functions (normalized with the theoretically best solution: $\min_t MSE(U, X(t))$) of the reconstructed image using different methods with Gaussian (a) and Poisson (b) noise; The proposed ADE based stopping condition gives a lower bound to any other methods.

DOI:10.15774/PPKE.ITK.2011.005



(a) The effect of the noise on the output quality of the different methods.



(b) The effect of inaccurate PSF estimation on the output quality of the different methods. The deviation of the deconvolution PSF's size from the blurring kernel's size is represented between -40% to +40%.

Figure 4.2: The stability of the methods for different noise levels and inaccurate estimation of the PSF. All curves are normalized with the maximum value of the baseline curve ($X_{(t=0)} = Y$).

DOI:10.15774/PPKE.ITK.2011.005

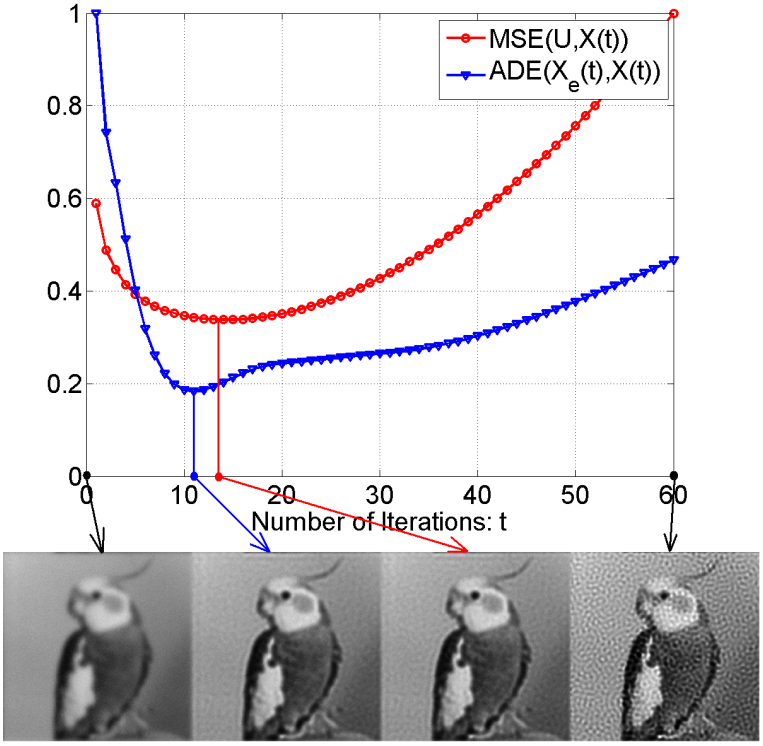


Figure 4.3: The estimation results by using the measurable $ADE(X_e(t), X(t))$ and the unmeasurable $MSE(U, X)$ functions. The proposed $ADE(X_e(t), X(t))$ function stops the deconvolution (at $t = 11$) close to the theoretically best iteration ($t = 14$). Both curves are normalized with their maximum value to be able to illustrate and compare their characteristics.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 5

Conclusions and Perspectives

The stopping condition is a common problem for non-regularized iterative deconvolution methods. In this part of the dissertation a novel method was described to automatically estimate the stopping condition based on the orthogonality of the change of the estimated signal and the signal itself at a given iteration. The proposed method provides an effective lower bound estimate to the conventional *ad-hoc* methods and experiments have proved its efficiency for different noise models and wide range of noise levels.

As future work we could use block based estimation of the optimal stopping condition to further enhance the results. It seems promising to try the algorithm with blind deconvolution methods. Deconvolution is a natural choice in situations where the acquired image is blurred. It is often used in astronomy and microscopy, and also it could be useful when motion blurred frames has to be restored in moving camera applications.

DOI:10.15774/PPKE.ITK.2011.005

Part II

Adaptive Image Decomposition into Cartoon and Texture Parts Optimized by the Orthogonality Criterion

DOI:10.15774/PPKE.ITK.2011.005

In this part a decomposition method is presented that splits the image into geometric (or cartoon) and texture parts. Following a total variation based preprocessing step, the core of the proposed method is an anisotropic diffusion with an orthogonality based parameter estimation and stopping condition. The quality criterion is defined by the theoretical assumption that the cartoon and the texture components of an image should be orthogonal to each other. The presented method has been compared to other decomposition algorithms through visual and numerical evaluation to prove its superiority.

The first chapter of this part presents the cartoon/texture decomposition problem along with the most important solutions from the literature. In Chapter 7 the proposed orthogonality based method is described. Numerical and visual results (Chapter 8) are followed by our conclusions in Chapter 9.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 6

Problem Formulation and Overview of Cartoon/Texture Decomposition Methods

Image decomposition into meaningful components has a key role in many image processing applications. By removing the noise [7,58], texture, reflections [59,60], fog/smoke [61] or shadows [62–64] and leaving only the main components of an image significantly helps the content understanding.

In this part, we focus on decomposition into texture and cartoon components. Texture is defined as small elementary pattern which is repeated periodically or quasi-periodically in space while under cartoon we mean the piecewise smooth geometrical components of the image. This kind of image decomposition can be useful for image compression where compressing the cartoon and the texture components separately can provide better results [17, 19], for image denoising [7, 58] since zero mean oscillatory noise can be regarded as a fine texture, image feature selection [6], 2D and 3D computer graphics and main edge detection as illustrated in [5], etc.

Recently published algorithms for texture/cartoon decomposition [5,6,8,9,65] are mostly based on Total Variation ($\boldsymbol{TV}$) minimization inspired by the work of Yves Meyer [66]. Total variation based regularization dates back to Tikhonov [23]. The most widely known form was introduced in image processing by Mumford and Shah [67] for image segmentation and later by Rudin *et al.* [7] for noise removal through the optimization of a cost function as follows:

$$\inf\left\{E_{TV}(u) = \int_{\Omega} |Du| + \lambda \int_{\Omega} v^2\right\} \tag{6.1}$$

where Ω is an open subset of $\mathbb{R}^2$, u is the cartoon component of the original image f, $v = f - u$ is the oscillatory or textured component, λ is a regularization parameter, and $\int_{\Omega} |Du|$ denotes the total variation of u in Ω, defined as follows:

$$\int_{\Omega} |Du| := \sup\left\{\int_{\Omega} u \mathrm{div}\phi(x)\mathrm{d}x : \phi \in C_0^1(\Omega, \mathbb{R}^2), \|\phi\|_{L^\infty} \leq 1\right\}, \tag{6.2}$$

DOI:10.15774/PPKE.ITK.2011.005

where $C_0^1(\Omega, \mathbb{R}^2)$ is the set of continuously differentiable vector functions of compact support contained in Ω, and $\|.\|_{L^\infty}$ is the essential supremum norm.

The first part produces a smooth image with bounded variation upon energy minimization for the cartoon component, while the second ensures that the result is close to the initial image. The regularization of Rudin *et al.* [7] (***ROF*** in the following) was used as an image denoising and deblurring method, since it removes fine, oscillating, noise-like patterns, but preserves sharp edges of the cartoon component.

In [66] Meyer proposed a different norm for the second, texture part of (6.1), which is better suited for oscillatory components than the standard L_2 norm:

$$\inf \int_\Omega |Du| + \lambda \|v\|_* \tag{6.3}$$

where $\|.\|_*$ is defined on a G Banach space as follows:

$$\|v\|_* = \inf_{g1,g2} \left\| \sqrt{g_1^2(x,y) + g_2^2(x,y)} \right\|_{L^\infty} \tag{6.4}$$

over all $g_1, g_2 \in L^\infty(\mathbb{R}^2)$ such that $v = \nabla \cdot \vec{g}$ where $\vec{g} = (g_1, g_2)$. In other words, the L^∞-norm must be minimized over all g_1 and g_2 such that $v(x,y) = \partial_x g_1(x,y) + \partial_y g_2(x,y)$. Other variations of (6.1) are summarized in [5].

There are methods which minimize the total variation in a wavelet framework [68, 69] for restoration of textured images. Wavelets are also used for carton/texture decomposition in [70], where they are combined with the the fundamental Rudin-Osher-Meyer Banach space decomposition. As we can see from these works wavelets are used for texture decomposition only in a combination with TV, since as stated in [71] TV minimization technique gives much better results in texture and noise decompositions than wavelets.

Beside variational methods Independent Component Analysis (***ICA***) [72, 73] and especially Morphological Component Analysis are also used (***MCA***) [74–76] for image decomposition. For cartoon/texture separation MCA was found to be more successful, which is a sparse-representation-based image decomposition method developed by Starck *et al.* in 2004 in a series of papers [74–76]. It has been shown that MCA can be used for separating the texture from the piecewise

DOI:10.15774/PPKE.ITK.2011.005

smooth component [75], for inpainting applications [76] or more generally for separating several components which have different morphologies. MCA has been extended to the multichannel case by Bobin *et al.* [77, 78]. The main idea behind MCA is to use the morphological diversity of the different features contained in the data, and to associate each morphology to a dictionary of atoms for which a fast transform is available.

In [8], the authors propose an image decomposition and texture segmentation method based on sparse representation and Principal Component Analysis (***DPCA***). They compared their method to [75] and found the results quite similar, though DPCA faster.

In [9], a TV-based algorithm (***DOSV***) is introduced to find the optimal value of the fidelity parameter λ (6.1) using the observation of Aujol *et al.* in [79] concerning the independence of the cartoon and texture image components.

Looking at the palette of the different solutions, we can see that the decomposition into cartoon and textured partitions requires tackling the following:

- Adaptive scale definition of texture and cartoon (or outline) details;
- Reasonable process that filters out textured parts while keeping the main outlines;
- Quality criterion for the efficiency of the decomposition: goal function of the process.

In the following, we overview the related contributions and then we introduce our proposed solutions to the above tasks.

6.1 Works Related to the Proposed Method

In this section we shortly summarize published results closely related to the proposed method: non-linear filtering is introduced in [5], Anisotropic Diffusion in [4, 80, 81] and measures of independence in [1, 79, 82].

DOI:10.15774/PPKE.ITK.2011.005

6.1.1 BLMV Nonlinear Filter

Buades *et al.* have recently proposed a non-linear method inspired by (6.3) (based on the names of the authors it will be called ***BLMV*** filter in the following) that calculates Local Total Variation (***LTV***) for each pixel on f before and after filtering the image with a σ-sized low pass filter, L_σ, inspired by Y. Meyer [66]. The LTV can be formalized as follows:

$$LTV_\sigma(f)(r) := L_\sigma * |Df|(r). \tag{6.5}$$

The *relative reduction rate* of the calculated LTVs shows if the observed pixel is part of the texture or the cartoon, since the LTV of the oscillatory parts will change radically, while the LTV of the cartoon parts will be left virtually unchanged (although blurred). The relative reduction rate of LTV is defined by a function $r \to \lambda_\sigma(r)$, given by

$$\lambda_\sigma(r) := \frac{LTV_\sigma(f)(r) - LTV_\sigma(L_\sigma f)(r)}{LTV_\sigma(f)(r)} \tag{6.6}$$

which gives us the local oscillatory behavior of the function f. The composition of the cartoon image u is based on this information:

$$\begin{aligned} u(r) &= w\left(\lambda_\sigma(r)\right)\left(LTV_\sigma * f\right)(r) + \left(1 - w(\lambda_\sigma(r))\right) f(r) \\ v(r) &= f(r) - u(r), \end{aligned} \tag{6.7}$$

where $w(.)$ function is defined as follows:

$$w(r) = \begin{cases} 0 & r < a_1 \\ (r - a_1)/(a_2 - a_1) & a_1 \le r \le a_2 \\ 1 & r > a_2 \end{cases} \tag{6.8}$$

with $a_1 = 0.25$ and $a_2 = 0.5$.

The results of this simple method are impressive on the presented examples in [5]: the edges are preserved as long as σ is not too large and the texture components are blurred with L_σ.

The right choice of σ is important to get the best result; however, it is possible

DOI:10.15774/PPKE.ITK.2011.005

that there is no such σ that eliminates all the textures but keeps the non-texture components on the cartoon. The existence of a content adaptive scaling parameter can be derived from scale-space theory, as has been introduced in the works of Lindeberg [83].

6.1.2 Anisotropic Diffusion

The general goal of diffusion algorithms is to remove noise from an image by using partial differential equations. Diffusion algorithms can be classified as isotropic or anisotropic. Isotropic diffusion can be described by the following equation:

$$\frac{\partial f(x,y,t)}{\partial t} = \nabla^2 \cdot f \tag{6.9}$$

where $f(x,y,t) : \mathbb{R}^2 \to \mathbb{R}^+$ is the image in the continuous domain, with (x,y) spatial coordinates, t an artificial time parameter and ∇f the image gradient. $f(x,y,0)$ is the original image. This diffusion is equivalent to using a Gaussian filter on the image, which blurs not only the noise or texture components, but the main edges as well.

Gábor [84] and later Perona and Malik [4] proposed anisotropic diffusion (***AD***) functions that, according to scale-space theory (see works of Florack [85] or Alvarez, Lions and Morel [86]) allows diffusion along the edges or in edge-free territories, but penalizes diffusion orthogonal to the edge direction:

$$\frac{\partial f(x,y,t)}{\partial t} = \nabla \cdot \left(g\left(\|\nabla f\| \right) \nabla f \right) \tag{6.10}$$

where $\|\nabla f\|$ is the magnitude of the gradient and $g(.)$ is the weighting function that controls diffusion along and across edges. The discretized form of their diffusion equation is as follows:

$$\begin{aligned} f(x,y,t+1) &= f(x,y,t) + \\ &+ \frac{\lambda}{|\eta(x,y)|} \sum_{(x',y') \in \eta(x,y)} \nabla^{(x',y')} \left(g\left(\left\| \nabla^{(x',y')} f(x,y,t) \right\| \right) \right) \nabla^{(x',y')} f(x,y,t) \end{aligned} \tag{6.11}$$

where I is the processed image, (x,y) is a pixel position, t now denotes discrete

DOI:10.15774/PPKE.ITK.2011.005

time steps (iterations). The constant $\lambda \in \mathbb{R}^+$ is a scalar that determines the rate of diffusion, $\eta(x, y)$ is the spatial neighborhood of (x, y), $|\eta(x, y)|$ is the number of neighboring pixels. $\nabla^{(x',y')} f(x, y, t)$ is an approximation of the image gradient at a particular direction:

$$\nabla^{(x',y')} f(x, y, t) = f(x', y', t) - f(x, y, t), (x', y') \in \eta(x, y) \tag{6.12}$$

AD belongs to a theoretically sound scale-space class of differential processes ensuring the denoising of an image along with the enhancement of its main structure [87]. We will show that the AD proposed by Perona and Malik [4] is not suitable for cartoon/texture decomposition, since the texture part might contain high magnitude edges, which would inhibit the diffusion. As a solution to this problem, Sprljan *et al.* [17] suggest that the AD algorithm is used with modified weights: instead of using $\left\|\nabla^{(x',y')} f(x, y, t)\right\|$ as the parameter of the weighting function, they use the edges of the Gaussian filtered image, $\nabla(G_\sigma * f)$:

$$\left\|\nabla^{(x',y')} \left(G_\sigma * f\right)(x', y', 0) - \left(G_\sigma * f\right)(x, y, 0)\right\|, (x', y') \in \eta(x, y) \tag{6.13}$$

where G_σ is a Gaussian filter with σ width and $*$ denotes spatial convolution. Using a blurred image to control diffusion directions will give better results: texture edges will not hinder the diffusion, but the strong main edges will do. Yet the quality of the solution relies heavily on the σ parameter: with small σ, some texture might remain on the cartoon, while with greater σ, some of the cartoon edges will disappear.

In Chapter 7 we will propose an algorithm that utilizes the smoothing property of AD while it helps preserving edges based on whether they belong to a cartoon or texture and not based on their local amplitude.

6.1.3 Use of Independence in Image Decomposition

The independence of the cartoon and texture/noise parts of the image was used in denoising, decomposition [79] and restoration [82] algorithms.

In [79], Aujol *et al.* propose the use of correlation between the cartoon and the oscillatory (noise, texture) components of a decomposition to estimate the

DOI:10.15774/PPKE.ITK.2011.005

regularization parameter λ. The assumption of their model is that these two components are uncorrelated, which makes intuitive sense (as stated in [9]), since every feature of an image should be considered as either a cartoon feature or a textural/noise feature, but not both. If a feature only appears in one of the two decomposition components, than there is no correlation between these components for the region consisting of all of the pixels in the feature.

In the previous Part, the ADE – introduced in [1] – was used as a measure of independence to automatically find the best stopping point for an iterative non-regularized image deconvolution method. As the deconvolution problem is ill-posed, after a certain point, further iterations will only amplify the noise on the estimated image. The heart of the method is to find the iteration where the change of the estimated image in one time step $X(t)-X(t-1)$ and the estimated image $X(t)$ are the most independent of each other. The described ADE measure (see (3.2)) is somewhat similar to correlation [79], but it is based on the orthogonality of two image partitions (e.g. clear image and noise). Comparing the ADE to the standard correlation – where zero-mean vectors are used to calculate the scalar product and the normalization is done with the standard deviation of the vectors – we can see that they are very similar: if both Q and P had zero mean, the two measures would actually give the same result. However, in cartoon texture decomposition only the texture part has an inherent zero mean, while the cartoon does not. This makes a small difference in the resulting decomposed images in favor of the ADE measure, as will be shown in Chapter 8; ADE strengthens the image partitions to being truly independent (geometrical orthogonality in $\mathbb{R}^n$), while *corr* is for the estimation of regression.

In the following we will show how independence can be used to separate better the texture and cartoon parts of the image by using the ADE orthogonality measure to locally estimate the best parameter of the BLMV filter. The edge inhibitions of the AD are first initialized by the filtered image. Then, the ADE is calculated again on the diffused image to stop the diffusion at the point where the orthogonality of cartoon and texture components is maximal.

To sum up, we offer theoretically clear solutions for the main issues:

- Adaptive scale definition by using locally optimal BLMV filter tuned by ADE measure;

DOI:10.15774/PPKE.ITK.2011.005

- Anisotropic Diffusion, initialized by the new adaptive BLMV to better separate texture from cartoon;
- Orthogonality criterion for the quality measure of the decomposition (stopping condition to AD).

In the following we overview in detail our proposed solutions for the above tasks. To validate the proposed method, we will show results on real life images (see Fig. 6.1), and also on artificial images where numerical evaluation is possible.

DOI:10.15774/PPKE.ITK.2011.005

(a) Barbara

(b) Geometry

(c) City

(d) Pillar

(e) Zebra

Figure 6.1: Images used for visual evaluation.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 7

Cartoon/Texture Decomposition Using Independence Measure

In this chapter, the orthogonality based cartoon/texture decomposition method is described in detail. The core algorithm is the AD, which is initialized and stopped using the BLMV filter and the ADE independence measure.

7.1 Locally Adaptive BLMV filter

As has been mentioned earlier, the BLMV filter uses the same σ-sized low pass filter for the whole image, even though there is no guarantee for the existence of a single σ that would remove all texture from the image without blurring the cartoon edges. Fig. 7.1 illustrates this problem: the same part of the image is filtered with two different σ values: $\sigma = 3$ pixels (pix in the following) in the first case and $\sigma = 4$ pixels in the second. With the smaller σ the filtered image contains unwanted texture components while with the larger σ important cartoon edges are blurred.

We propose the use of different σ for the different parts of the image based on the independence of the removed texture component and the remaining cartoon component. This theory is similar to the one proposed in [79], although in our case the parameter selection has to be locally adaptive. The reason for this difference lies in the purpose of the methods: while in [79] the goal of the authors was noise removal, where one can assume that the parameters of the noise are the same for the whole image, here we want to remove texture components which may vary in many aspects (e.g. scale, magnitude) across the image.

To make the filter locally adaptive, BLMV filtered images were calculated for a given range of the scale parameter: $\sigma_i \in [s_1, s_2]$. Let $u_{\sigma_i}, v_{\sigma_i}$ denote the cartoon and texture components of the f input image, produced by the BLMV filter with σ_i parameter. The image is then divided into non-overlapping small cells (5 pixel by 5 pixel in our experiments), and around each cell a larger block (21 by 21) is

DOI:10.15774/PPKE.ITK.2011.005

centered, in which the ADE measure is calculated:

$$ADE(u_{\sigma_i}^{(x,y)}, v_{\sigma_i}^{(x,y)}) = \left| \arcsin \left(\frac{\langle u_{\sigma_i}^b(x,y) v_{\sigma_i}^b(x,y) \rangle}{|u_{\sigma_i}^b(x,y)| \cdot |v_{\sigma_i}^b(x,y)|} \right) \right|, \tag{7.1}$$

where $u^b(x,y)$ and $v^b(x,y)$ denote the cartoon and texture components of the *block* which is centered around the cell containing (x,y) pixel.

It is worth noting that the texture component of an image should have zero mean, since it is the difference of the textured area and the diffused background. To eliminate the consequences of the quantization error through the iterations, the texture component is biased to be zero-mean when the ADE function is computed.

The σ with minimal ADE is chosen to be the parameter for each pixel in the cell. For the output cartoon image the value of the pixel $u_a(x,y)$ will be the following:

$$u_a(x,y) = u_{\sigma_m^{(x,y)}}(x,y) \tag{7.2}$$

$$\sigma_m^{(x,y)} = \underset{\sigma_i \in [s_1, s_2]}{\arg\min} \left(ADE \left(u_{\sigma_i}^b(x,y), v_{\sigma_i}^b(x,y) \right) \right) \tag{7.3}$$

This cell-based scheme is used to reduce the computational workload: instead of calculating the block correlation for each pixel, we calculate it for small cells. To avoid the blocking effect, a soft Gaussian smoothing was used on the parameter image of the same size as the input image and containing the corresponding σ value in each (x,y) point: $p(x,y) = \sigma_m^{(x,y)}$. An example result of the described method can be seen on Fig. 7.2. and the corresponding parameter image on Fig. 7.3.

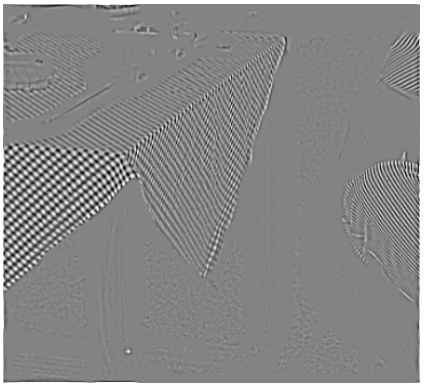
7.2 Anisotropic Diffusion with an Adaptive BLMV Filter and ADE Stopping Condition

The above described adaptive BLMV filter (***aBLMV*** in the following) clearly performs better than the original one (see Chapter 8), but it still faces a problem at the borders where cartoon and texture parts meet: either the cartoon edges are blurred, or the texture remains on the cartoon component close to cartoon edges.

DOI:10.15774/PPKE.ITK.2011.005

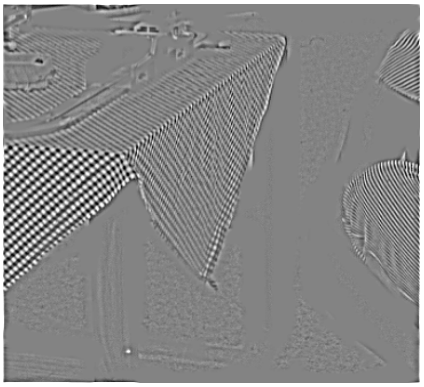
(a) Original Image (Part of the Barbara image)

(b) Cartoon component with $\sigma = 3pix$



(c) Texture component with $\sigma = 3pix$

(d) Cartoon component with $\sigma = 4pix$



(e) Texture component with $\sigma = 4pix$

Figure 7.1: The cartoon and texture component of a part of the Barbara image produced by the BLMV method with $\sigma = 3pix$ and $\sigma = 4pix$, respectively. Note that the texture of the tablecloth (on the left side of the image) is not completely removed by the smaller sigma, while the edges of the cover are blurred if we choose a larger sigma that eliminates the texture from the cover.

DOI:10.15774/PPKE.ITK.2011.005

Figure 7.2: Cartoon and texture components of the BLMV filtered Barbara image (Fig. 6.1(a)) with adaptive selection of the σ parameter.

DOI:10.15774/PPKE.ITK.2011.005

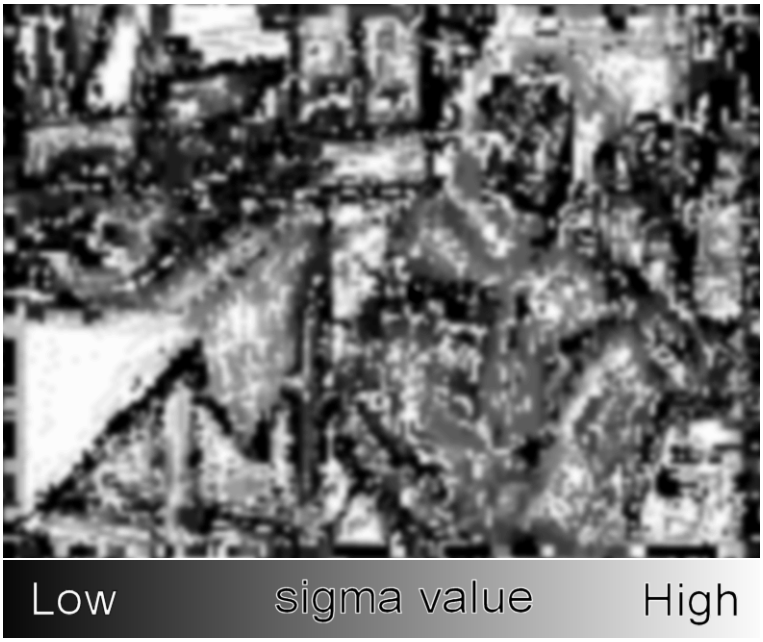


Figure 7.3: The parameter map of Barbara image (Fig. 6.1(a)). The brighter the pixel on the map the greater the σ value used on that image part. In this image the value of σ is between 0 pix and 5 pix.

We propose to use AD initialized with a cartoon image produced by aBLMV filter and stopped by ADE measure. AD preserves high magnitude edges and blurs weaker ones, but obviously a texture can contain strong edges while a cartoon edge can be weak. As a result, AD may blur important edges of the cartoon and keep unwanted edges of the texture.

Similarly to [17], where the diffusion weight function was calculated on a Gaussian blurred version of the image, we propose to calculate the weight function $g(.)$ of (6.11) by using the aBLMV-filtered image resulting in the following diffusion equation:

$$f(x,y,t+1) = f(x,y,t) + \\ + \frac{\lambda}{|\eta(x,y)|} \sum_{(x',y')\in\eta(x,y)} \nabla^{(x',y')} \left(g\left(\nabla^{(x',y')} u_a(x,y) \right) \right) \nabla^{(x',y')} f(x,y,t) \tag{7.4}$$

Note that the aBLMV filter, could be easily replaced in the algorithm by

DOI:10.15774/PPKE.ITK.2011.005

any other method, which blurs the texture but preserves cartoon edges. We tried various methods, like simple Gaussian blur or the linear filter used in [5], and we found that the aBLMV filter performs the best. TV based methods like ***TVL1*** [6] and ROF [7] were also tested, but the results of the combined methods (TVL1+AD+ADE or ROF+AD+ADE) were no better than the results of the respective method's (TVL1 or ROF) alone.

On u_a of (7.2), the texture parts are blurred and they do not contain strong edges, while the cartoon parts are more or less preserved. Choosing a low value for the rate of diffusion λ means that the diffusion can preserve even the weak edges of the cartoon part, but it blurs texture parts completely (since it is not inhibited by edges). Fig. 7.4 shows the cartoon and texture components produced by the method proposed above (AD with ADE).

To avoid oversmoothing of important edges, the iteration of the AD must be stopped at the right moment. For this purpose, we utilize the independence property of cartoon and texture components in the same manner as we did in Section 7.1, with the difference that here we are searching for the iteration count i that minimizes $ADE(u_i, v_i)$ for each block.

The cartoon component of the proposed method is produced as follows:

$$u(x,y) = f(x,y,t_{ADE}), \tag{7.5}$$

$$t_{ADE} = \underset{i=1..I_{max}}{\arg\min}(ADE(f^b(x,y,i), v^b(x,y,i))) \tag{7.6}$$

where $f(x,y,t_{ADE})$ is the (x,y) pixel of the diffused image after t_{ADE} iterations, I_{max} is the maximum number of diffusion iterations, $f^b(x,y,i)$ and $v^b(x,y,i) = f^b(x,y,0) - f^b(x,y,i)$ are the cartoon and texture components, respectively, of the *block* that is centered around the cell containing (x,y) pixel after the ith diffusion iteration.

If the diffusion is not stopped automatically, but after a fixed number of iterations, then some parts of the cartoon component will be apparent on the texture image, as it can be seen on Fig. 7.5.

DOI:10.15774/PPKE.ITK.2011.005

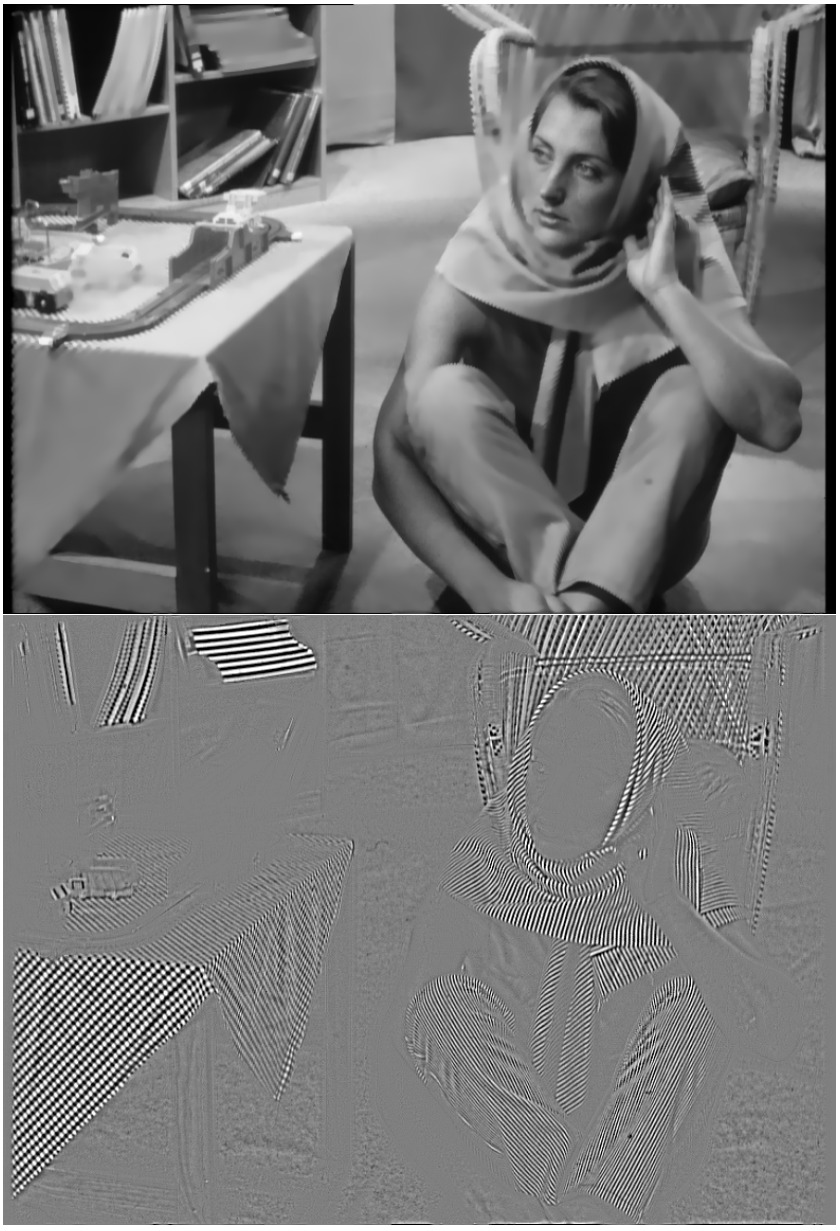


Figure 7.4: The cartoon and texture component of the Barbara image produced by the proposed anisotropic diffusion model with ADE based stopping condition.

DOI:10.15774/PPKE.ITK.2011.005

Figure 7.5: The cartoon and texture component of the Barbara image produced by the proposed anisotropic diffusion model after 100 iterations.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 8

Results

The evaluation of the quality of cartoon/texture decomposition is usually done on visual examples, since there is no generally accepted objective method for ground truth generation in the case of real images. Sometimes it is difficult even for a human to decide if a certain part of the image is texture or not.

Hence, to evaluate the quality of the different methods, we show the decomposition results of example images (see Fig. 6.1), but we also evaluate numerically the competing methods on artificial images (see Fig. 8.6) where the ground truth cartoon and texture parts are available.

We have compared the proposed method to the following decomposition methods: BLMV-filter [5], aBLMV-filter (we proposed in this work), Anisotropic Diffusion [4], DPCA [8], DOSV [9], ROF [7], TVL1 [6]. The codes for the above methods were provided by the authors, and we used them with the best tuned parameters in each individual test case. For numerical evaluation, we used the parameters that gave the best numbers, and in the case of subjective evaluation, the parameters that gave the best visual result. For the proposed method, we kept all the parameters, except one: the σ range for the aBLMV was changed to the same transparent scale parameter as it was for the original BLMV. The other parameters were set to a constant value: the maximum number of iterations for the AD was set to 100, the λ scalar parameter of (6.11), which controls the rate of diffusion was set to 2. Note that $\lambda = 2$ makes the effect of the diffusion 75% weaker than the standard diffusion (without any weighting), making the AD very sensitive to edge inhibitions, which helps the better preservation the cartoon edges. The only parameter that was not constant during the tests is the $[s_1, s_2]$ range of the σ. The usual values were $s_1 = 0.5pix$ and $s_2 = 7pix$. The only time when the values were different was for the City image, where we set $s_2 = 4pix$ to preserve better the cartoon details of the image.

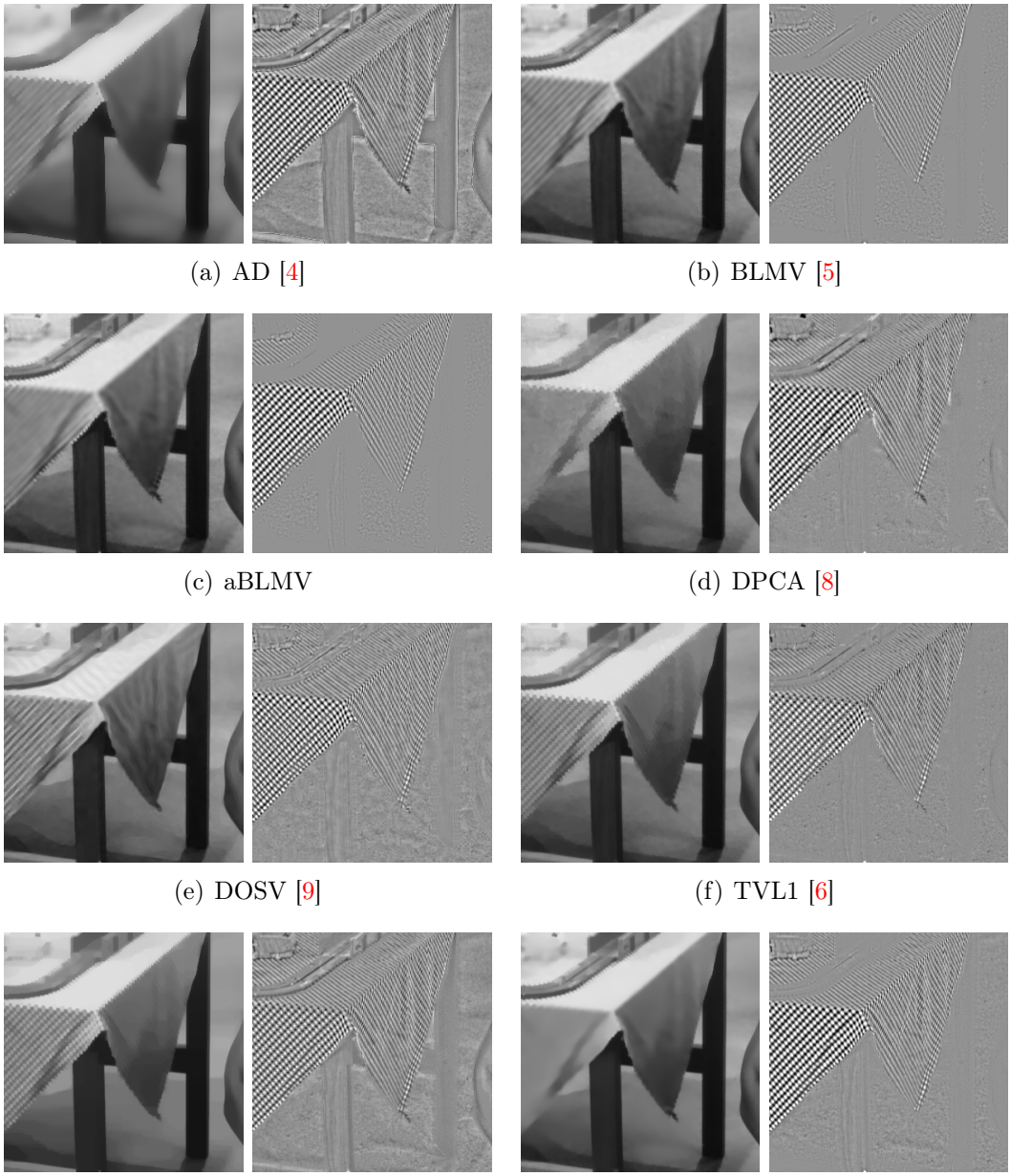
For better visibility, the texture images' contrast was linearly stretched on the demonstrated figures.

DOI:10.15774/PPKE.ITK.2011.005

8.1 Visual Evaluation

For the visual evaluation, one has to consider how strong the remaining cartoon parts on the texture image and the remaining texture part on the cartoon image are. For a part of the Barbara image, we can see on Fig. 8.1 that 5 methods (AD, BLMV, TVL1, ROF, DOSV) cannot completely eliminate the texture from the table cover, while there are cartoon edges apparent on the texture image. DPCA can eliminate the texture from the cartoon image, but the image itself becomes less smooth, and the slow changes of gray level values are also apparent on the texture image. The BLMV with adaptive local parameter selection (aBLMV) and the proposed method eliminate the texture from the cartoon while virtually no cartoon appear on the texture image (see Fig. 8.1). On the Geometry image, all the methods eliminate the texture from the cartoon part, but all of them bring some cartoon edges on the texture (see Fig. 8.2). Here one should consider how strong the cartoon edges on the texture image are, and also how precise the cartoon part is. The third image shows city towers. This image has precise edges, which favors the TV based methods, especially ROF (see Fig. 8.3). However, some artifacts can be seen on ROF's cartoon image, as the rectangularly shaped cloud at the top of the building on the left, or the disappearing top of the same skyscraper. Most of the methods blur some parts of the image, and almost none of them can eliminate the vertical line texture from the darkest building. For the fourth image (Pillars), the question is how well the pillars are preserved on the cartoon image (or how strong the edges of the pillars on the texture image are) and how blurred the greenery in the background is (see Fig. 8.4). Here we can say that BLMV, DOSV and TVL1 produce good results, but they are outperformed by aBLMV and the proposed method, while AD and ROF preforms very poorly: the texture is slightly blurred on the cartoon image and the edges of the pillar are already obviously present on the texture part. DPCA blurs the texture the best (similarly to the proposed method), but, in the meantime, it brings some strong cartoon edges to the texture part. The last image, the Zebra is quite challenging, since the texture of the Zebra has a wide range of sizes. See results on Fig. 8.5. AD and ROF preform poorly, since most of the texture remains on the cartoon, while non-textural parts like slow changes of gray level values and non-textural

DOI:10.15774/PPKE.ITK.2011.005



(a) AD [4] (b) BLMV [5]

(c) aBLMV (d) DPCA [8]

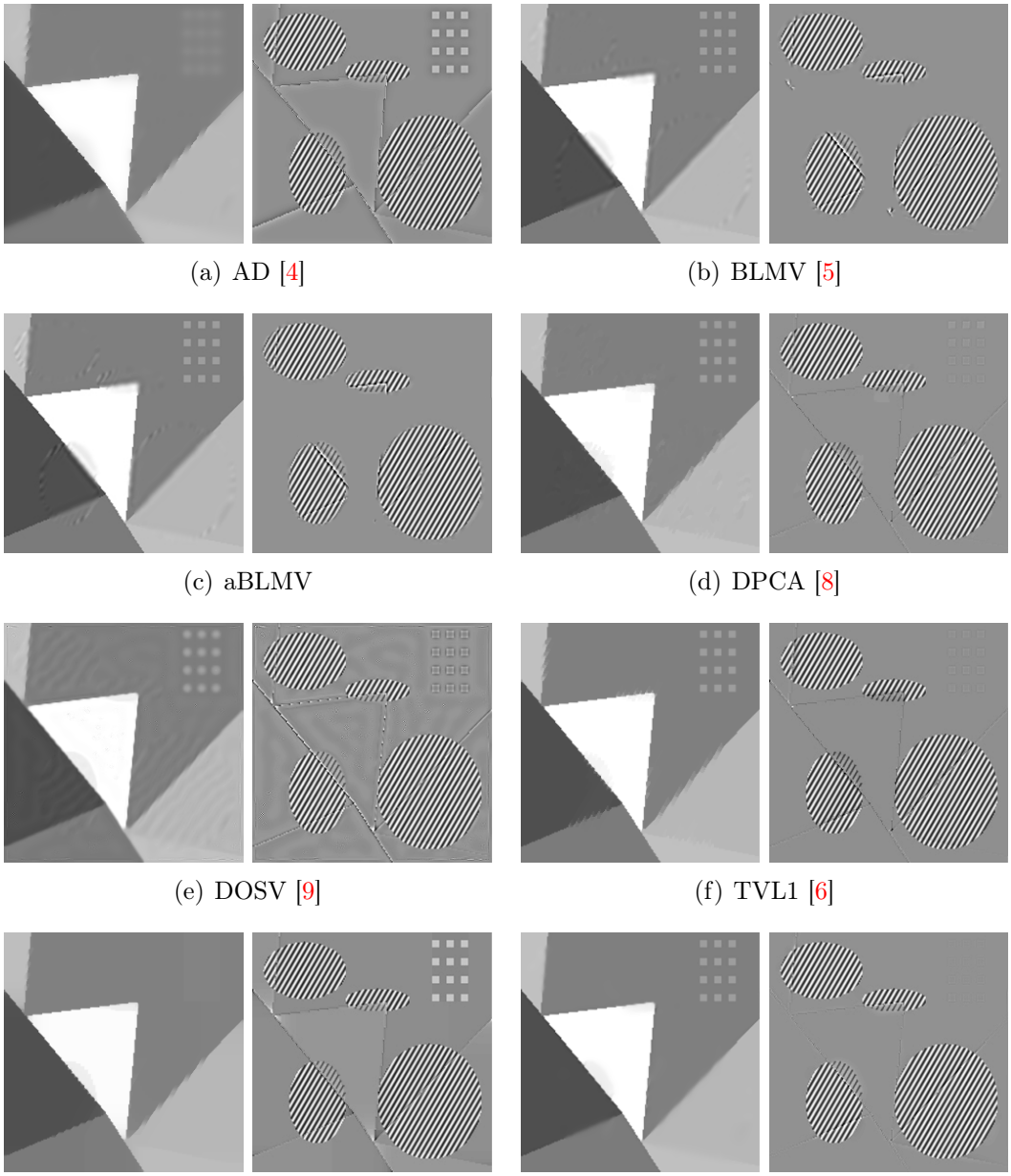
(e) DOSV [9] (f) TVL1 [6]

(g) ROF [7] (h) Proposed AD-aBLMV-ADE

Figure 8.1: Separation of cartoon and texture components (Barbara)

parts of the background are apparent on the texture image. As it was mentioned earlier, AD is not suited for the tasks, since the edges of the texture are stronger than some of the cartoon edges, therefore the cartoon edges are eliminated while

DOI:10.15774/PPKE.ITK.2011.005



(a) AD [4] (b) BLMV [5]

(c) aBLMV (d) DPCA [8]

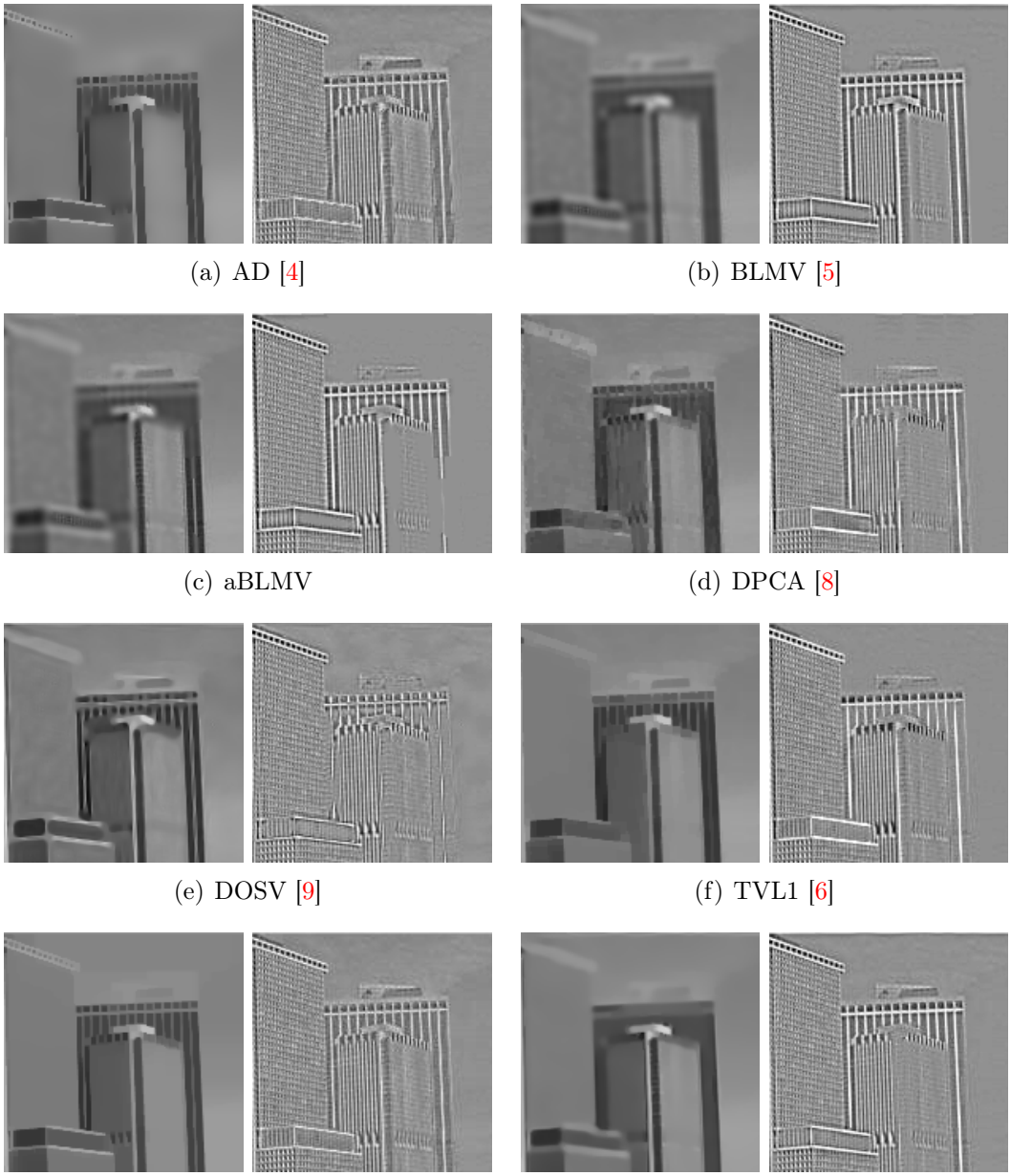
(e) DOSV [9] (f) TVL1 [6]

(g) ROF [7] (h) Proposed AD-aBLMV-ADE

Figure 8.2: Separation of cartoon and texture components (Geometry)

the texture edges are kept unchanged. DOSV cannot eliminate the larger texture parts without blurring the cartoon. BLMV blurs the cartoon even more, but it eliminates most of the texture, although not as efficiently as aBLMV or the

DOI:10.15774/PPKE.ITK.2011.005



(a) AD [4] (b) BLMV [5]

(c) aBLMV (d) DPCA [8]

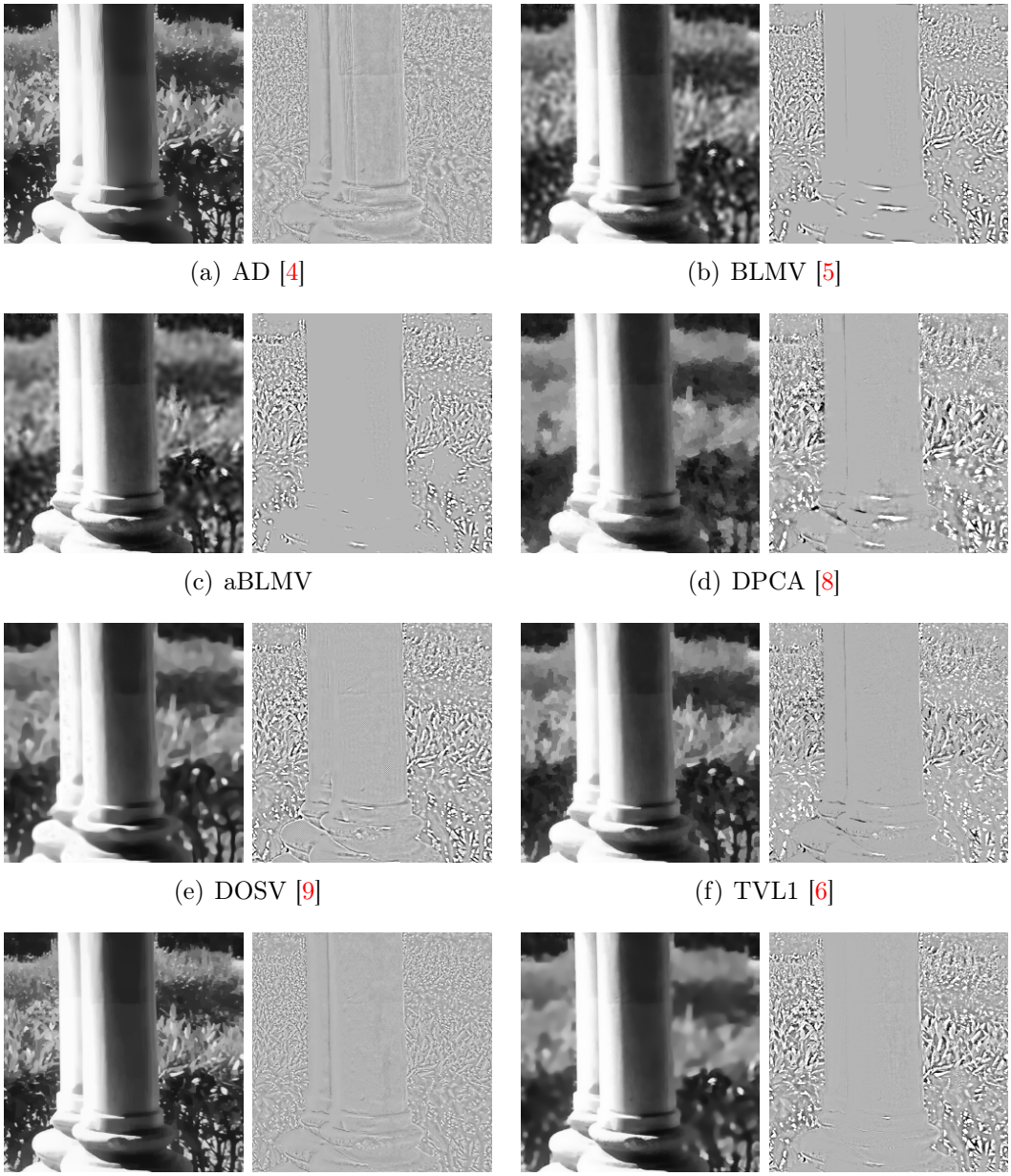
(e) DOSV [9] (f) TVL1 [6]

(g) ROF [7] (h) Proposed AD-aBLMV-ADE

Figure 8.3: Separation of cartoon and texture components (City towers)

proposed method. TVL1 and DPCA perform similarly: they both eliminate most of the texture, but a lot of non-textural edges are apparent on the texture image as well.

DOI:10.15774/PPKE.ITK.2011.005



(a) AD [4] (b) BLMV [5]

(c) aBLMV (d) DPCA [8]

(e) DOSV [9] (f) TVL1 [6]

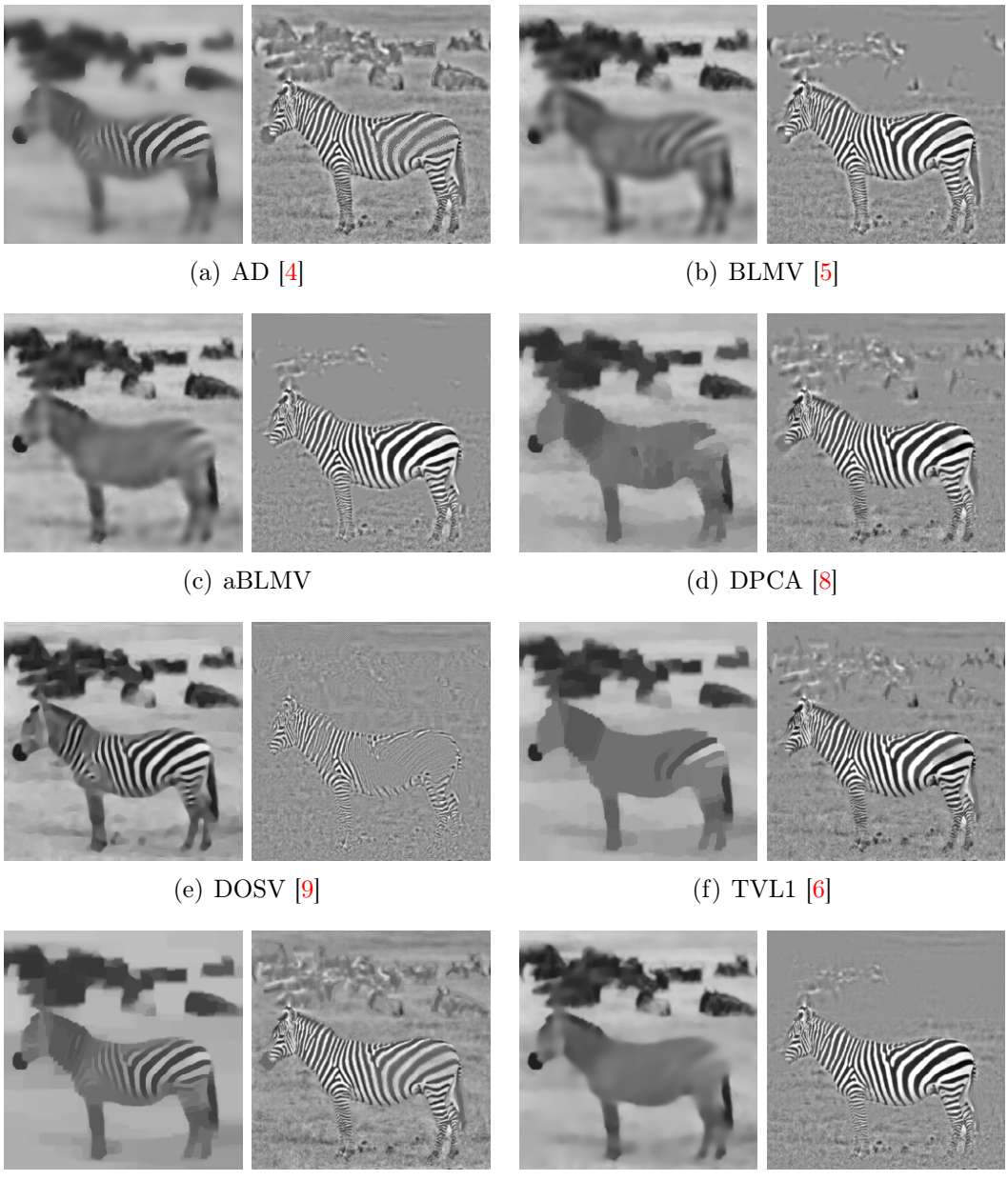
(g) ROF [7] (h) Proposed AD-aBLMV-ADE

Figure 8.4: Separation of cartoon and texture components (Pillar)

8.2 Numerical Evaluation

Numerical evaluation is a difficult task for cartoon/texture decomposition since usually there is no ground truth (GT) for the images. For this reason most

DOI:10.15774/PPKE.ITK.2011.005



(a) AD [4] (b) BLMV [5]

(c) aBLMV (d) DPCA [8]

(e) DOSV [9] (f) TVL1 [6]

(g) ROF [7] (h) Proposed AD-aBLMV-ADE

Figure 8.5: Separation of cartoon and texture components (Zebra)

DOI:10.15774/PPKE.ITK.2011.005

papers in the field lack this kind of comparison and rely only on subjective visual evaluation. We used artificial images for numerical evaluation where the ground truth is available.

The following measures were calculated to compare quality: edge absolute difference of the cartoon ($ead(u)$) and texture ($ead(v)$) images, the absolute difference of the cartoon images ($ad(u)$), and the correlation coefficient of the estimated texture and the GT texture. We define these measures as follows:

$$\begin{aligned} ead(u) =& |e(u') - e(u)|, \\ ead(v) =& |e(v') - e(v)|, \\ ad(u) =& |u' - u|, \\ corr(v', v) =& \frac{cov(v', v)}{\sigma_{v'}\sigma_v}, \end{aligned} \tag{8.1}$$

where u and v are the ground truth cartoon and texture images, u', v' are the cartoon and texture images produced by a decomposition method and $e(.)$ is the Prewitt edge image. For $ead(u)$, $ead(v)$ and $ad(u)$, the lower value means better result, while for the correlation coefficient, the higher values correspond to better results. In general the proposed AD-aBLMV-ADE method performs better than the rest (see TABLE 8.1-8.4).

	Scores for the 1st image of Fig. 8.6			
	ead(u)	ead(v)	ad(u)	corr(v',v)
AD [4]	**0.4585**	**0.3870**	3.7430	0.9460
BLMV [5]	0.8105	0.6698	4.5584	0.9238
aBLMV	0.7668	0.6333	4.3541	0.9298
DPCA [8]	0.8322	0.6701	5.7892	0.8939
DOSV [9]	0.5556	0.5130	3.7948	0.9422
TVL1 [6]	0.9311	0.7802	7.9885	0.7893
ROF [7]	0.6683	0.5237	4.6568	0.9367
AD-aBLMV-ADE	0.5075	0.4532	**3.6115**	**0.9479**

Table 8.1: Numerical results for the 1st image of Fig. 8.6. The best results are highlighted in bold.

DOI:10.15774/PPKE.ITK.2011.005

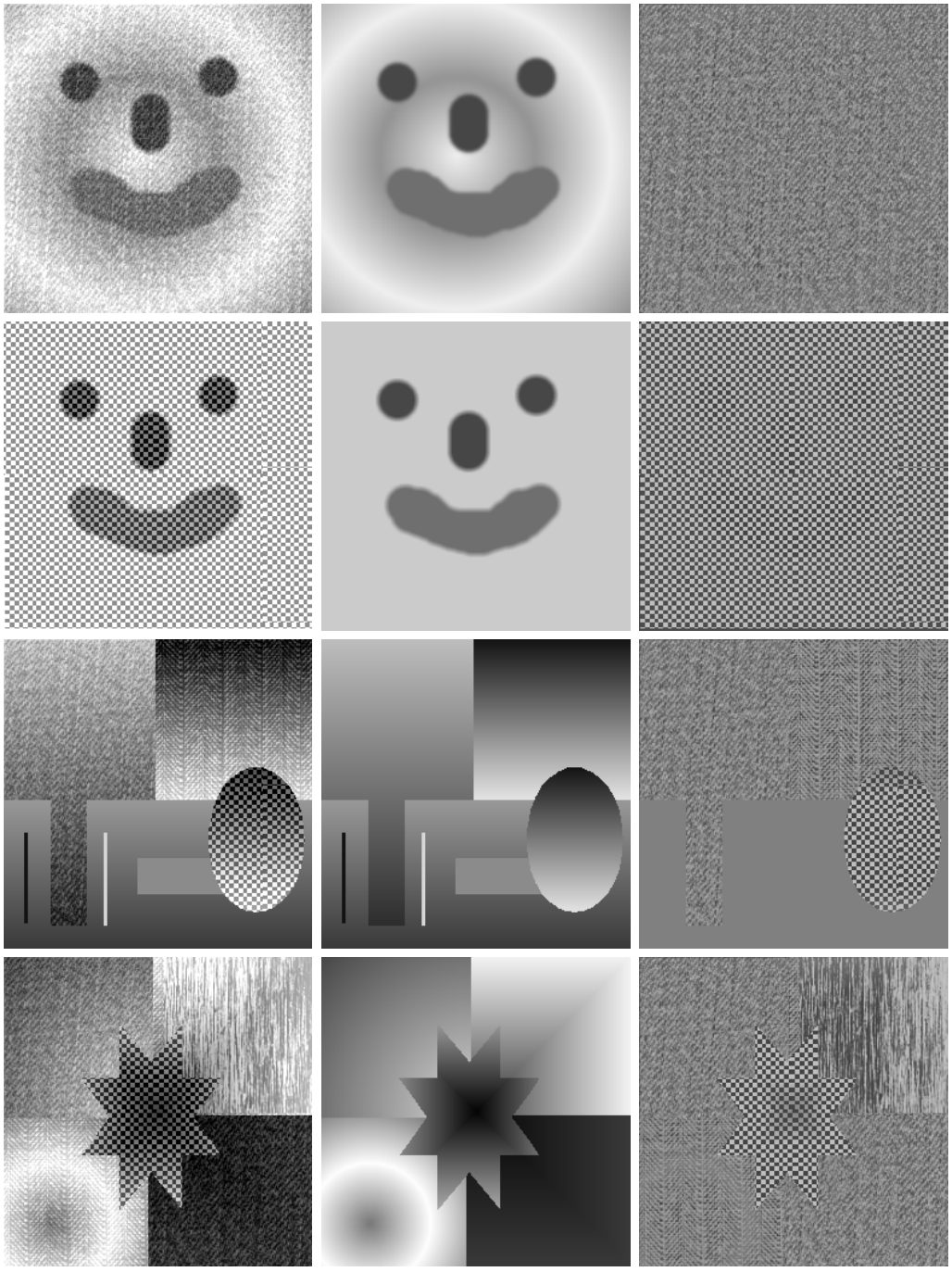


Figure 8.6: The artificial images and the corresponding ground truth components used for numerical evaluation. Left column: original image, Middle column: cartoon component, Right column: texture component.

DOI:10.15774/PPKE.ITK.2011.005

	Scores for the 2nd image of Fig. 8.6			
	ead(u)	ead(v)	ad(u)	corr(v',v)
AD [4]	0.5631	0.5373	2.1536	0.9564
BLMV [5]	0.8393	0.7611	2.6356	0.9263
aBLMV	0.7922	0.7182	2.4525	0.9311
DPCA [8]	0.7420	0.6596	2.8667	0.9361
DOSV [9]	0.4137	0.3911	1.5184	0.9780
TVL1 [6]	0.4339	0.3288	3.8850	0.8693
ROF [7]	0.5926	0.4855	2.6941	0.9508
AD-aBLMV-ADE	**0.2782**	**0.2408**	**1.0342**	**0.9862**

Table 8.2: Numerical results for the 2nd image of Fig. 8.6. The best results are highlighted in bold.

	Scores for the 3rd image of Fig. 8.6			
	ead(u)	ead(v)	ad(u)	corr(v',v)
AD [4]	0.3279	0.2386	1.7500	0.9678
BLMV [5]	0.4363	0.3229	1.8498	0.9661
aBLMV	0.3703	0.3037	1.5932	0.9811
DPCA [8]	0.5535	0.4322	2.3272	0.9658
DOSV [9]	0.3217	0.2788	1.6535	0.9826
TVL1 [6]	0.5589	0.4391	2.2239	0.9681
ROF [7]	0.5105	0.4064	2.3239	0.9658
AD-aBLMV-ADE	**0.2682**	**0.2385**	**1.2826**	**0.9876**

Table 8.3: Numerical results for the 3rd image of Fig. 8.6. The best results are highlighted in bold.

We have also compared the proposed method using ADE to the case when correlation is used instead as an independence measure. It shows that the pure orthogonality criterion performs slightly better than the correlation based comparison (see TABLE 8.5). As a drawback of the proposed method, we have to mention that the block based computation of ADE might cause problems at the

DOI:10.15774/PPKE.ITK.2011.005

	Scores for the 4th image of Fig. 8.6			
	ead(u)	ead(v)	ad(u)	corr(v',v)
AD [4]	0.4019	0.3268	1.6902	0.9949
BLMV [5]	0.4479	0.3147	1.4710	0.9974
aBLMV	0.3979	0.2805	1.4539	0.9974
DPCA [8]	0.6842	0.5437	9.2985	0.9118
DOSV [9]	0.3564	0.3090	1.2509	0.9984
TVL1 [6]	0.6917	0.4942	29.2620	0.7246
ROF [7]	0.2829	0.2846	2.3199	0.9926
AD-aBLMV-ADE	**0.1610**	**0.1304**	**1.2245**	**0.9985**

Table 8.4: Numerical results for the 4th image of Fig. 8.6. The best results are highlighted in bold.

Comparison of independence measures			
ead(u)	ead(v)	ad(u)	corr(v',v)
97.34%	97.31%	98.73%	100.06%

Table 8.5: Ratio of the error rates and the correlation of ADE based vs. Correlation based calculus. The results obtained by ADE are better than the ones obtained by correlation: the absolute differences have decreased while the correlation coefficient has slightly increased.

borders and might oversmooth small and weak cartoon parts, as it can be seen on the City image, where the top of the building is hardly recognizable. The computational time is also increased compared to the fast BLMV, TVL1 and ROF methods, but it is faster than DPCA or DOSV (see TABLE 8.6).

Computational Time	
AD [4]	3.3
BLMV [5]	0.90
aBLMV	8.8
DPCA [8]	271.5
DOSV [9]	710.7
TVL1 [6]	0.47
ROF [7]	1.3
AD-aBLMV-ADE	21.3

Table 8.6: Computational time (in seconds) of the different methods for the City image (436x232) on a Pentium IV 2 GHz notebook with 3GB memory.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 9

Conclusion and Perspectives

In this part novel and theoretically sound solutions were presented for the main issues of cartoon/texture decomposition, using anisotropic diffusion with ADE based iteration stopping. To initialize the diffusion inhibitions of AD, we used a BLMV nonlinear filter with adaptive parameter selection based on ADE calculus. Numerical results and visual comparisons show that the proposed method works with high efficiency and in quality it outperforms other algorithms introduced in recent years.

As it was mentioned in the introduction cartoon/texture separation has a lot of useful applications in image compression, 3D graphics, denoising, etc. The presented work (similarly to the deconvolution) could also be useful as a preprocessing step before motion estimation where the noise often causes false results.

DOI:10.15774/PPKE.ITK.2011.005

Part III

Detection of Moving Foreground Objects in Video Recordings with Strong Camera Motion

DOI:10.15774/PPKE.ITK.2011.005

This part describes a method for moving foreground object extraction in sequences taken by a wearable camera, with strong motion. Camera motion compensated frame differencing is enhanced with a novel kernel-based estimation of the probability density function of background pixels. The probability density functions are applied for filtering false foreground pixels on the motion compensated difference frame. The estimation is based on a limited number of measurements; therefore, a special, spatial-temporal sample point selection and an adaptive thresholding method is introduced to deal with this challenge. Foreground objects are built from detected foreground pixels using a robust clustering algorithm.

In the next chapter, an overview of existing foreground/background separation methods is given, followed by our motivation, the specific description of the problems we had to face and the presentation of the mathematical tools we applied to solve them. Chapter 11 describes the method we propose for foreground object detection in wearable video recordings; in Chapter 12, the experiments are presented which we performed to measure the quality of our method and compare it to another similar algorithm. At the end of this part Chapter 13 summarizes our experiences.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 10

Motivation and Problem Formulation

Wearable video capture has been recently gaining popularity due to the availability of new low-weight and low-energy consumption hardware. From the pioneering works of Steve Mann [88] in the domain of wearable computing, who worked at concealing image acquisition and computing power inside non invasive clothing, the technology has evolved to allow autonomous devices with a long battery life and image capture capabilities. One example is the SenseCam device [16], which can be hung around the neck due to its low weight while recording images all day long. This type of device produces a new kind of video data, which brings new possibilities from the point of view of recording and using data acquired during everyday activities [89]. The advent of personal video capture using video cameras or mobile phones with cameras is one of the factors leading to a sharp increase of the quantity and ubiquity of such data.

Automatic analysis approaches, which are working well for more traditional types of video, such as static, or motion controlled cameras, now need to be adapted, in order to be able to automatically extract meaningful information from these new data. Segmenting foreground objects from the background is one such basic module, which is of great interest, as it is commonly used to bootstrap many higher-level analysis algorithms such as object-of-interest detection and tracking. Strong motion and parallax, as well as low signal quality (caused by motion blur) make such videos very challenging. Moving object detection in such videos is not just a matter of camera motion compensation and then detection of foreground pixels as if it was a still-camera video. The main reason for this – beside the above mentioned difficulties – is the scene, which is always changing, hence there is not enough data to build the background model used in the case of still cameras.

In the following we will present how the problem of foreground detection evolved from simple constant background subtraction to foreground detection on recordings taken by moving cameras.

DOI:10.15774/PPKE.ITK.2011.005

10.1 Overview of Foreground/Background Separation Methods

The idea of background/foreground separation is not new, it has been an active field of research since the advent of digital videos [13, 15, 90–102]. It is computationally demanding, therefore its development is affected greatly by the evolution of technology through the increase of available computational power.

Generally there is no *a priori* information about the foreground, therefore it cannot be modeled directly, therefore the static background is modeled by using the information of consecutive frames from the past.

The simplest method of segmentation is Frame Differencing (***FD***). To use this technique we have to assume that the moving objects are always in motion, since if they stop the difference frame or foreground mask will be empty. The foreground mask in case of FD is the absolute difference of the actual and the previous frame:

$$F(t) = |I(t) - I(t-1)| \tag{10.1}$$

where $F(t)$ is the foreground mask, $I(t)$ is the actual frame at time instance t, and $I(t-1)$ is the previous frame. In [103] FD is used with post-processing techniques for moving object detection in an outdoor environment. Obviously the foreground mask produced with FD is not perfect, since it has non-zero values both at the current and the previous position of the moving object and possibly false negative detections (zero-valued pixels) at the overlapping positions.

To reduce the false detection, instead of calculating the difference of two consecutive frames a background image is calculated as the average or the median [63, 104] of the previous n frames and the foreground is generated by subtracting the background from the current frame. This method still have problem with changes of the background, slowly moving foreground objects, and so on.

To handle these difficulties, in [105] the background was statistically modeled by Gaussian distributions. The probability distribution function of each (x, y) pixel was approximated with a Gaussian distribution with μ mean and σ variation,

DOI:10.15774/PPKE.ITK.2011.005

based on the values that the pixel had on the past frames:

$$\begin{aligned}\mu(x,y,t+1) &= \alpha f(x,y,t) + (1-\alpha)\mu(x,y,t)\\ \sigma^2(x,y,t+1) &= \alpha(f(x,y,t)-\mu(x,y,t))^2 + (1-\alpha)\sigma^2(x,y,t),\end{aligned} \tag{10.2}$$

where $f(x,y,t)$ is the pixel value at (x,y) position at time t and α is the learning rate. This model can handle slow changes of the background as the change of the pixel value is slowly incorporated into the model. Each pixel of a new frame then can be classified as either part of the background or part of the foreground based on how it fits to the background model:

$$|f(x,y,t)-\mu(x,y,t)| > \tau(x,y,t) \tag{10.3}$$

where τ is a threshold usually depending on σ.

Using only one Gaussian for a pixel to model the background has its limitations: it cannot handle changes in the background like shadows cast by clouds, doors in open or closed state, etc. In [15, 106] Stauffer and Grimson present a method that assigns a mixture of Gaussians to a pixel, and in this way more than one kind of background can be handled. The number of modes is predefined (usually between 3 and 5) and each mode has 3 parameters: μ and σ are the mean and variance, while ω is the weight of the Gaussian. At every new frame, some of the Gaussians will be updated based on their distance from the current pixel value: if

$$|f(x,y,t)-\mu(x,y,t)| < 2.5\cdot\sigma_i(x,y,t), \tag{10.4}$$

then the mean and variance of the ith mode of the (x,y) pixel is updated by the running average (10.2). The modes of a pixel are ranked based on the values of ω_i/σ_i. The valid background mode is always the first in the ranking. The weights of the modes are updated and normalized in each time instance. This method is generally referred to as the Gaussian Mixture Model (***GMM***) or Mixture of Gaussians (***MoG***). To further improve this idea recursive equations are used in [90–92] to constantly update the parameters and also to simultaneously select the appropriate number of components in the GMM for each pixel.

Carminati*et al.* [93] suggest a more elegant solution for choosing the appropri-

DOI:10.15774/PPKE.ITK.2011.005

ate component of the mixture for a given pixel instead of (10.4), which consists in likelihood maximization. For a given sample of a pixel $f(x, y, t)$ they maximize the likelihood of a Gaussian component η_i in the following mixture:

$$P(f(x,y,t)) = \sum_{i=1}^{K} \omega_i(x,y,t) \cdot \eta\left(f(x,y,t), \mu_i(x,y,t), \sigma_i^2(x,y,t)\right), \tag{10.5}$$

where K is the number of the components. $P(f(x, y, t))$ expresses the probability of observing $f(x, y, t)$ value at the (x, y) position at time t. According to the Bayes theorem the conditional likelihood of the ith Gaussian in the mixture for a sample $f(x, y, t)$ will be

$$l(f(x,y,t)) = \frac{\omega_i \eta_i}{\sum_{k=1}^{K} \omega_k \eta_k}. \tag{10.6}$$

For a given sample the selection of the best Gaussian $\Theta_i^* = (\omega_i^*, \mu_i^*, \sigma_i^{2*})$ in the mixture is done with the following log-likelihood maximization:

$$\Theta_i^* = \underset{\Theta_i=(\omega_i,\mu_i,\sigma_i^2)}{\arg\max} \left(\log \omega_i - \left(\frac{\log \sigma_i^2}{2} + \frac{(f(x,y) - \mu_i)^2}{2\sigma_i^2} \right) \right). \tag{10.7}$$

Nevertheless high computational cost of (10.8) which is due to log-computation is penalizing for real time applications. Hence a simplifications were proposed: assuming that all the weights of the Gaussians in the mixture are equal (10.8) can be rewritten as follows:

$$\Theta_i^* = \underset{\Theta_i=(\mu_i,\sigma_i^2)}{\arg\min} \left(\frac{\log \sigma_i^2}{2} + \frac{(f(x,y) - \mu_i)^2}{2\sigma_i^2} \right). \tag{10.8}$$

For spatial-temporal coherence of the detection Carminati *et al.* [93] also proposed a Markov Random Field-based (***MRF***) regularization.

Another algorithm [107] based on [15, 106] integrates an adaptive GMM with a support vector machine classifier to detect and segment moving objects in dynamic backgrounds for video surveillance.

Beside the GMM, which is a parametric probabilistic model, there are also non-parametric models like [108, 109], that use kernel density estimation as background model. It can handle situations where the background of the scene is clut-

DOI:10.15774/PPKE.ITK.2011.005

tered and not completely static but contains small motions such as tree branches and bushes. These periodic changes of the background represent significant difficulties in the case of surveillance cameras. The authors of [13] propose a method that can deal with periodically changing background elements by modeling the dynamic characteristics using the optical flow parameters in a higher dimensional space as a feature. The background model is calculated by kernel density estimation with data-dependent bandwidth.

The cost of flexibility of the kernel density estimation methods is the increased computational complexity with higher memory requirements. The method described in [110] tries to solve this problem by using sequential kernel density estimation with mean shift algorithm for initial mode selection.

In [95] a fast algorithm for background modeling and subtraction was proposed. Sample background values at each pixel were quantized into codebooks which represent a compressed form of background model for a long image sequence. This allows to capture structural background variation due to periodic-like motion over a long period of time under limited memory. This method can handle scenes containing moving backgrounds or illumination variations (shadows and highlights).

All these research works deal with a stationary camera, when the background on 2D sequenced scene is static. The extraction of moving objects in the case of a moving camera is even more challenging. There are two different motions in an observed scene: the egomotion of the camera and the motion of the object. To extract the motion of the object, the camera motion has to be estimated and compensated [96].

GMM-based methods are used with non-static cameras also. In [111] the authors present a Bayesian approach for simultaneously detecting the moving objects and estimating their motion in image sequences taken with a moving camera mounted on the top of a mobile robot. To model the background, the algorithm uses the GMM approach combined with the motion cue in a maximum *a posteriori* probability-MRF framework.

In [97] the authors present a surveillance system with a moving camera. Its motion is estimated with feature tracking. Moving object detection is achieved via background compensation. Here the camera is an outdoor surveillance camera,

DOI:10.15774/PPKE.ITK.2011.005

with more planar view and less abrupt camera motion, than in the wearable case.

In [98] a system is introduced which uses a single camera to extract human motion in an outdoor environment. The camera is installed on a mobile robot and the motion of the camera is compensated using corresponding feature sets and outliers detection. The positions of moving objects are estimated using an adaptive particle filter and Expectation-Maximization (EM) algorithm.

A similar system is described in [99], where the authors propose an integrated computer vision system designed to track multiple humans and extract their silhouette with a pan-tilt stereo camera. The detection of foreground objects is performed by camera motion compensation and disparity segmentation.

The authors of [100] propose a framework, derived from a perceptual grouping principle, namely the Helmholtz principle. This principle states that perceptually relevant events are perceived because they deviate from a model of complete randomness. Detection is then said to be performed *a contrario*: moving regions appear as low probability events in a model corresponding to the absence of moving objects in the scene. However in the presence of strong parallax some parts of the static background may be considered as a moving object.

The method described in [101] deals with detection of motion regions in video sequences observed by a moving camera, in the presence of strong parallax, due to 3D static objects. The proposed method classifies each image pixel into planar background, parallax or moving regions using 2D planar homographies, an epipolar constraint and a so called structure consistency constraint. The method was tested on different outdoor sequences with encouraging results, but a known limitation of the algorithm is that it cannot handle abrupt camera motion, which is a requirement in our case.

In [102] the background is modeled by one single probability density function using a nonparametric density estimation method over a joint domain-range representation of image pixels. The foreground is also modeled based on previous detections and used competitively with the background model. The strength of the method is its capability to handle dynamic textures, cyclic motions and "nominal" camera motion. They use static cameras, where the camera motion comes from the effect of wind or trembling of the ground. The magnitude of these motions can be very strong but the scene does not change and the number

DOI:10.15774/PPKE.ITK.2011.005

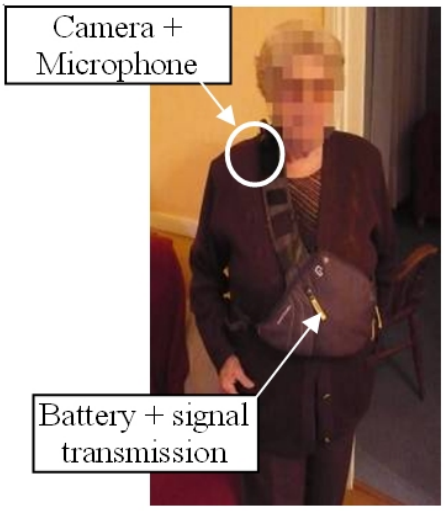
of measurements is not limited as strictly as in our case.

In the following sections the application that motivated our work will be presented along with the challenges we had to face.

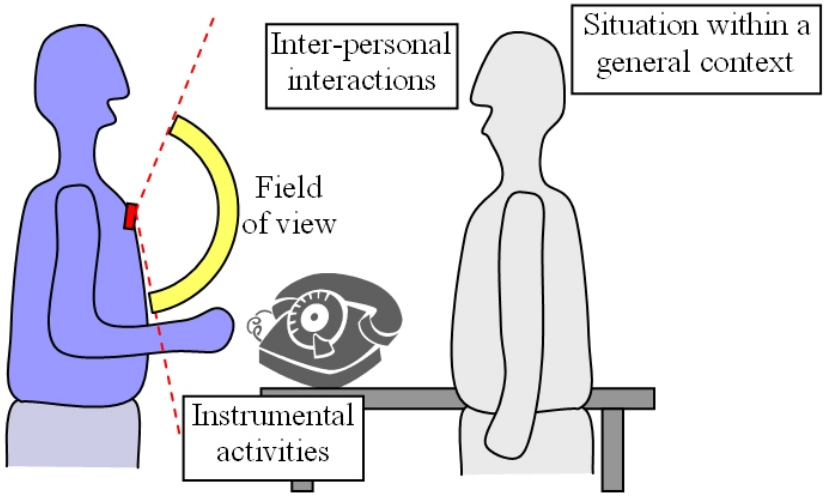
10.2 Motivation

The work described in this part is motivated by the development of new methods for the observation of patients suffering from dementia diseases such as Alzheimer's. The observation of patients during their daily activities helps to diagnose dementia stages and propose targeted assistance. Such observations at home are not much developed at the moment, because of the tremendous amount of time that it would require to be generalized. The stakes are high, as the recent PAQUID epidemiological study [112] has shown that the presence of some restrictions in Instrumental Activities of Daily Living (***IADL***) was correlated with the future appearance of a dementia related disease. The IADLs are here considered to be the events of interest. They correspond to the interaction of the patient with objects during daily activities, such as preparing the meal and having dinner, washing dishes, receiving a phone call, opening doors, etc. Difficulties can arise at several cognitive levels, from the ability to control one's hands from a motor point of view to the elaboration and the correct realization of strategies to accomplish the activities. Monitoring such difficulties requires acquiring enough pertinent information, which motivated the development of a wearable video capture device we first introduced in [113], and which is represented in Fig. 10.1. In this device the camera is worn close to the shoulder of the patient. Two types of camera can be used: a wide-angle camera (with approximately 100° angle of view) and a standard button camera (with approximately 40° angle of view). Some examples of image snapshots acquired with such a device are shown in Fig. 10.2. Wide-angle cameras proved to be more useful as they allow for better recording of events close to the camera such as IADLs, but the recorded video is difficult to analyze as the image undergoes a strong non-linear deformation. The button cameras allow for a good understanding of the environment as well as the analysis of instrumental activities. This work was only concerned with button camera recordings.

DOI:10.15774/PPKE.ITK.2011.005



(a) The wearable device we used for the experiments: the camera is placed on the shoulder of the patient, while the battery and the transmission device are in a small bag.



(b) The figure shows an illustration of the video acquisition: the goal is to record situations (personal interactions, instrumental activities), which can help the practitioners to make a diagnosis.

Figure 10.1: The acquisition device and context.

DOI:10.15774/PPKE.ITK.2011.005

Figure 10.2: Examples of image snapshots from acquired videos with wide-angle camera (top line) and button camera (bottom line).

10.3 Problem Formulation

The objective of video analysis in the current work (illustrated in Fig. 10.3) is to extract meaningful events related to the IADLs in order to provide practitioners with video indexing assistance when using the videos for diagnostic purposes. The goal of this work is to help finding the meaningful events in the long video sequences so that the practitioners can provide the diagnosis based on them. The automatic analysis of the retrieved parts for medical diagnosis is not part of the current task. We can identify two important low-level cues that are useful for IADL related event analysis. The segmentation of the hand of the patient and the detection of persons moving in front of the camera are strongly related to the instrumental activity or the situation of the patient, which helps understanding the context of an action. These two segmentations both benefit from a low-level segmentation that could separate the moving foreground (corresponding to hands or persons) from the background, which remains static in the 3D world. One challenge of such a task resides in the relative instability of the camera position, which is strongly coupled to the movement of the wearer and the low quality of the frames due to motion blur. In the following section the mathematical tools

DOI:10.15774/PPKE.ITK.2011.005

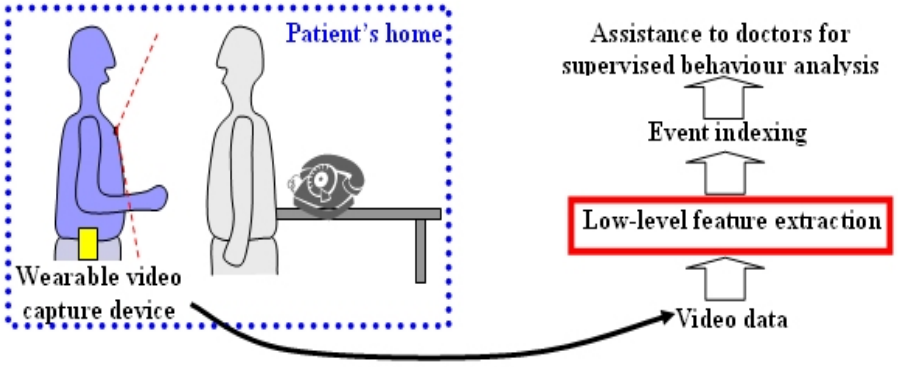


Figure 10.3: Principle of multi-level analysis for the video acquired using the wearable camera.

will be presented that we utilized to overcome these difficulties.

10.4 The Applied Mathematical Techniques

10.4.1 Kernel Density Estimation Methods

Density estimation is the construction of an estimate of an unobservable probability density function based on observed data. The data are usually thought of as random samples drawn from an unobservable density function. The density estimation can be parametric or non parametric.

In the case of parametric density estimation one assumes that the data are drawn from a specific functional form which depends on a few parameters, for example a Gaussian distribution with mean μ and variance σ^2. The density f underlying the data could then be estimated by finding estimates of the parameters from the data. Perhaps the most popular form for parametric density estimation in image processing is the GMM [15, 106].

Non-parametric estimation of a density function makes less rigid assumptions about the distribution of the observed data. The most frequently used non-parametric density estimators are the histogram, nearest neighbor methods, and kernel-based estimators. For a detailed description of these and other estimation methods see [114].

In the following we will go into the details of Kernel Density Estimators

DOI:10.15774/PPKE.ITK.2011.005

(***KDE***) which were first introduced by Fix and Hodges in 1951 [115]. Later Emanuel Parzen [12] and Murray Rosenblatt [116] independently created their current form, known also as the Parzen-Rosenblatt window. The aim of this estimation method is to extrapolate the measured data and build a regular density function. Kernel functions are used for the extrapolation, which are placed at each measurement point. Let $v_1, v_2, ..., v_n$ be a set of d-dimensional, i.i.d. sample points in $\mathbb{R}^d$, drawn from a random variable that follows a probability density function f. Let $K : \mathbb{R}^d \to \mathbb{R}$ be a kernel satisfying the following conditions:

$$\int_{\mathbb{R}^d} K(v)\mathrm{d}v = 1 \tag{10.9}$$

$$\int_{\mathbb{R}^d} vK(v)\mathrm{d}v = 0 \tag{10.10}$$

$$\int_{\mathbb{R}^d} vv^T K(v)\mathrm{d}v = I_d \tag{10.11}$$

The equations (10.9-10.11) together with non-negativity define K as a zero-mean, identity covariance Probability Density Function (***PDF***). We can define the kernel-based approximation of function f at the estimation point v for a given n as follows:

$$\tilde{f}(v) = \frac{1}{n\|H\|^{\frac{1}{2}}} \sum_{i=1}^{n} K\left(H^{-1}(v - v_i)\right) \tag{10.12}$$

where n is the number of v_i observed data points or samples, and H is a smoothing parameter (bandwidth matrix). The bandwidth of the kernel is a free parameter which determines its width and height, and exhibits a strong influence on the resulting estimate. For the sake of simplicity let us consider a diagonal bandwidth matrix:

$$H = \begin{bmatrix} \sigma_1^2 & ... & 0 \\ 0 & ... & 0 \\ 0 & ... & \sigma_d^2 \end{bmatrix} \tag{10.13}$$

where each σ_i^2 represents the bandwidth for a dimension i.

Fix and Hodges introduced this estimator for the univariate case with fixed bandwidth and uniform kernel in 1951 [115]. Rosenblatt [116] and Parzen [12] studied the general class of univariate fixed bandwidth kernel estimators.

DOI:10.15774/PPKE.ITK.2011.005

The selection of the kernel function and especially the bandwidth parameter are obviously very important [117], since they both have a strong influence on the accuracy and the smoothness of the PDF estimate.

10.4.2 Selection of Bandwidth and Kernel Function

The bandwidth controls the degree of smoothing. A too small value of the bandwidth will result in an undersmoothed estimate with fake peaks, while a too large value can eliminate important details of the estimated PDF.

The kernel bandwidth can either be fixed or varying. Choosing a fixed bandwidth means that H is constant for all samples at each measurement point. This is a simple and computationally efficient method, but in many situations it is less accurate than the variable bandwidth method. It has problems with sparse data: for instance at the "tail" of univariate PDFs as shown by Silverman for suicide data [114] or with multivariate data where the higher dimensionality results in a sparse representation [118]. It is also less accurate for multi-modal densities as stated in [119].

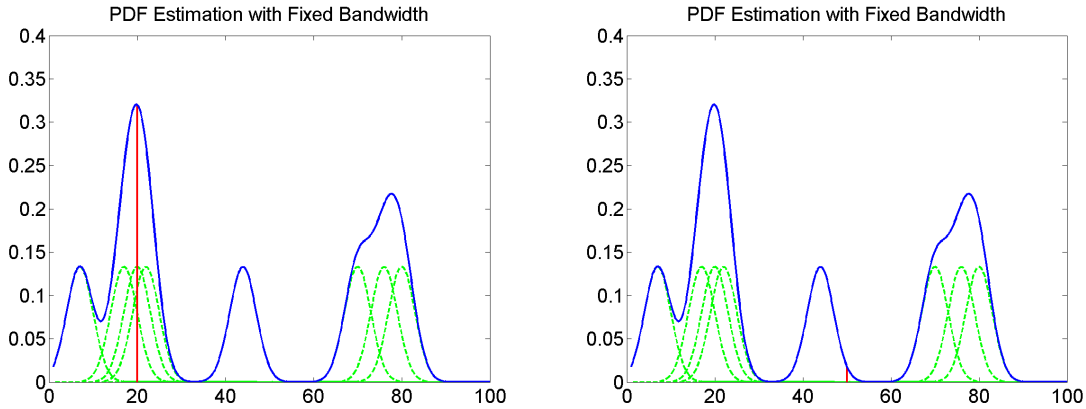
To adapt the bandwidth to the sample data, let us consider two traditional ways for variable bandwidth selection: the *balloon estimator* [118, 120–122] and the *sample-point estimator* [123, 124]. The difference between the two lies in how the bandwidth is varied.

In the case of the balloon estimator a different but fixed bandwidth is selected for each estimation point [125], hence H is a function of the estimation point: $H = H(v)$. In other words, the kernel width is determined by the sample density around the estimation point. The density estimation function with balloon estimation is given by

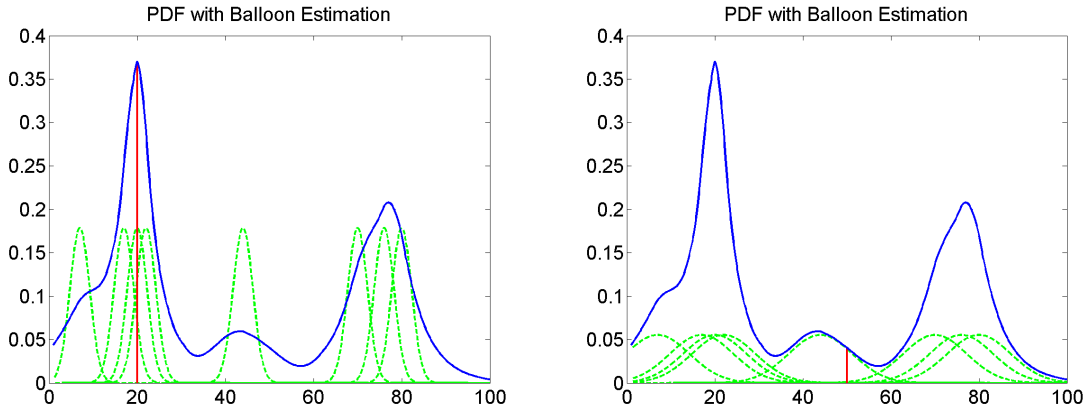
$$f_s(v) = \frac{1}{n}\sum_{i=1}^{n} K_{H(v)}(v - v_i) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{\|H(v)\|^{\frac{1}{2}}} K\left(H^{-1}(v)(v - v_i)\right), \qquad (10.14)$$

where n is the number of measurements, and $H(v)$ is the balloon estimated bandwidth matrix at v estimation point. Taken pointwise, balloon estimation behaves exactly as fixed bandwidth estimation. The biggest drawback of balloon estimators is that when considered as a global estimate it fails to integrate to 1 [125].

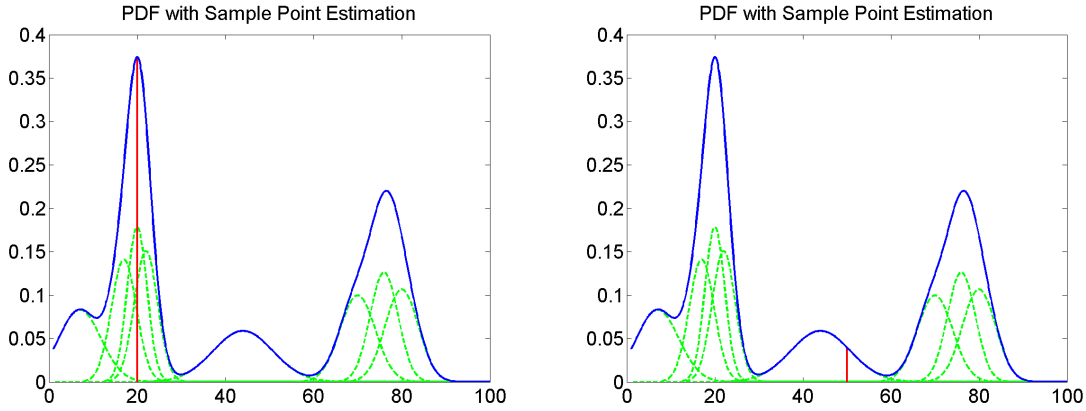
DOI:10.15774/PPKE.ITK.2011.005



(a) Fixed Kernel Estimation



(b) Balloon Estimation



(c) Sample Point Estimation

Figure 10.4: Examples for the different bandwidth selection methods. The kernels are with dashed green lines, the estimated PDF is with solid blue line and the vertical red line signs the estimation point.

DOI:10.15774/PPKE.ITK.2011.005

Sample point estimators, on the other hand, are themselves densities as long as the kernel function is also a density.

For sample point estimation the kernel bandwidth is a function $H(v_i)$ of the sample point v_i on which it is centered, thus all the kernels building up the density may have different bandwidth. A sample-point density estimator can be defined as follows:

$$f_s(v) = \frac{1}{n}\sum_{i=1}^{n} K_{H(v_i)}(v - v_i) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{\|H(v_i)\|^{\frac{1}{2}}} K\left(H^{-1}(v_i)(v - v_i)\right) \qquad (10.15)$$

where n is the number of measurements, and $H(v_i)$ is the sample-point bandwidth matrix associated to the ith sample point v_i. Sample point estimators can suffer from "non-locality", which means that data very far away from the estimation point may have a strong effect on the estimate.

The difference between the fixed, the balloon and the sample point bandwidth estimators are visualized on Fig. 10.4.

Name	Kernel Function
Uniform	$1/2$
Triangle	$1 - \|x\|$
Epanechnikov	$\frac{3}{4}(1 - x^2)$
Quadratic	$\frac{15}{16}(1 - x^2)$
Tricube	$(1 - x^3)^3$
Gaussian	$\frac{1}{2\pi} e^{(-\frac{1}{2}x^2)}$

Table 10.1: Kernel functions for probability density estimation. For all functions except for Gaussian $|x| \leq 1$.

It is generally accepted that the choice of the bandwidth is more important than the choice of the kernel function [117, 126]. Several kernel functions can be considered for density estimation e.g.: Epanechnikov, Gaussian, uniform, triangle, quadratic, etc.). See Table 10.1 for a list of the mathematical definition of these functions.

DOI:10.15774/PPKE.ITK.2011.005

10.4.3 Clustering Methods

Clustering is a part of unsupervised learning techniques where the aim is to find hidden structure in unlabeled data. Since the samples given to the learner are unlabeled, there is no error or reward at the evaluation of a potential solution. This distinguishes unsupervised learning from supervised learning and reinforcement learning. The clustering approaches as given by Jain *et al.* [127] are the following:

- Hierarchical Clustering Algorithms;
- Partitional Algorithms;
- Mixture-Resolving and Mode-Seeking Algorithms;
- Nearest Neighbor Clustering;
- Fuzzy Clustering;

Hierarchical Clustering Algorithms find successive clusters using previously established ones. They are either agglomerative or divisive. Agglomerative algorithms begin with each element as a separate cluster and merge them into successively larger clusters, while divisive algorithms begin with the whole set and proceed to divide it into successively smaller clusters. Most hierarchical clustering algorithms are variants of the single-link [128], complete-link [129], and minimum-variance [130, 131] algorithms.

Partitional Algorithms typically determine all clusters at the same time, but they can also be used in a divisive way. The k-means is one of the simplest and most commonly used algorithms [132].

For *Mixture-Resolving and Mode-Seeking Algorithms* the underlying assumption is that the patterns to be clustered are drawn from one of several distributions, and the goal is to identify the parameters of each and their number. Mostly it is assumed that the individual components of the mixture density are Gaussian. Traditional approaches to this problem involve obtaining a maximum likelihood estimate of the parameter vectors of the component densities [133]. Expectation Maximization (EM) algorithm has been applied to the problem of parameter estimation [134].

DOI:10.15774/PPKE.ITK.2011.005

Nearest Neighbor Clustering [135] is an iterative procedure: it assigns each unlabeled pattern to the cluster of its nearest labeled neighbor pattern, provided the distance to that labeled neighbor is below a threshold. The process continues until all patterns are labeled or no additional labeling occurs.

Traditional hard clustering approaches generate disjoint partitions, where each pattern belongs to one and only one cluster. *Fuzzy clustering* associates each pattern with every cluster using a membership function [136]. The output of such algorithms is a clustering, but not a partition.

Probably the most common algorithms are k-means [132] and ***DBSCAN*** (Density-Based Spatial Clustering of Applications with Noise) [14], which is a density-based hierarchical algorithm. K-means is a simple partitional algorithm,which iteratively partitions the data points into a predefined k number of clusters. Given an initial k means of the clusters, the algorithm alternates between two steps:

(I) Assignment step: each data point is assigned to the cluster with the nearest mean.

(II) Update step: calculate the new means to be the centroid of the data point in the cluster.

This method is simple and fast, but it requires *a priori* knowledge of the number of clusters, is biased to produce circular shaped clusters, and the result is highly dependent on the initial cluster means. To avoid the sometimes poor clusterings found by the standard k-means algorithm the authors in [137] propose an algorithm for choosing the initial values for the k-means clustering.

DBSCAN overcomes the limitation of k-means since it does not require *a priori* knowledge of the number of clusters and it can produce arbitrary-shaped clusters.

DBSCAN requires two parameters: ϵ, the maximum distance of two neighboring points and the minimum number of points required to form a cluster. It starts with an arbitrary starting point that has not been visited. If the ϵ-neighborhood of one point contains a sufficient number of points, a cluster is formed. Otherwise, the point is labeled as noise. Noise points might later be in the range of a sufficiently sized ϵ-environment of a different point and hence be made part of a cluster.

DOI:10.15774/PPKE.ITK.2011.005

Figure 10.5: Examples of clusters discovered by DBSCAN. Unlike the clusters of k-means, these clusters are not biased to be circular-shaped. [14]

If a point is found to be part of a cluster, its ϵ-neighborhood is also part of that cluster, as is their own ϵ-neighborhood, and so on. This process continues until the cluster is completely found. Then, a new unvisited point is retrieved and processed until there is no further unlabeled data point. The disadvantage of this method is that it cannot handle high dimensionality or large differences in cluster densities, and its run time complexity is $O(n^2)$, where n is the number of data points.

10.4.4 Global Motion Estimation

Moving object detection can be divided into two classes based on how they model the scene: 2D algorithms [138, 139], which are applied when the scene can be approximated by a flat surface and 3D algorithms [140–142], which model the scene in 3D, but they only work well when certain parameters, such as the focus of expansion (foe) are available or can be precisely estimated [143]. In our case the foe is not available *a priori* and its estimation would be biased by the moving objects, hence we applied the less complicated 2D model even though the scene is not perfectly planar. If the scene is assumed to be planar then the camera motion can be estimated with the following model:

$$x' = \frac{a_1 + a_2 x + a_3 y}{a_7 x + a_7 y + 1}, y' = \frac{a_4 + a_5 x + a_6 y}{a_7 x + a_7 y + 1} \tag{10.16}$$

where $a_1, ...a_8$ are the motion parameters, (x, y) denotes the spatial coordinates of a pixel in the current frame and (x', y') denotes the coordinates of the cor-

DOI:10.15774/PPKE.ITK.2011.005

responding pixel in the previous frame. The choice of an appropriate camera motion model depends on the degrees of freedom of camera motion. For arbitrary rotation and zoom, the above described 8-parameter perspective motion model (10.16) is used. Various motion models can be derived from this model. In this work we used a simple affine model assuming $a_7 = a_8 = 0$.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 11

Moving Foreground Object Detection

11.1 General Scheme of the Proposed Method

The method we propose consists of 3 steps: (1) Motion Compensated Frame Differencing, (2) Estimation of Foreground Filter Model, (3) Detection of Moving Objects (see Fig. 11.1). As the name implies the compensation of camera motion occurs in step (1). After compensation the two frames have the same coordinate system and an error image can be calculated as a difference of compensated frames. This error image should contain only the foreground regions. Due to changes in perspective, quantization error and other sources of noise (transmission noise, strong motion blur, etc.), the foreground contains a lot of false positives. To eliminate this noise we use a foreground filter model in step (2), built on a so-called Modified Error Image (***MEI***) resulting from step (1). Because of the movement of the person who is wearing the camera and changes of capturing conditions in a natural environment, this model has to be continuously updated. The last step is the detection of the moving objects. Based on the model from step (2), the background pixels are eliminated from the MEI and then a density-based clustering (DBSCAN), as described in the previous chapter, is applied to the remaining foreground pixels to build foreground objects.

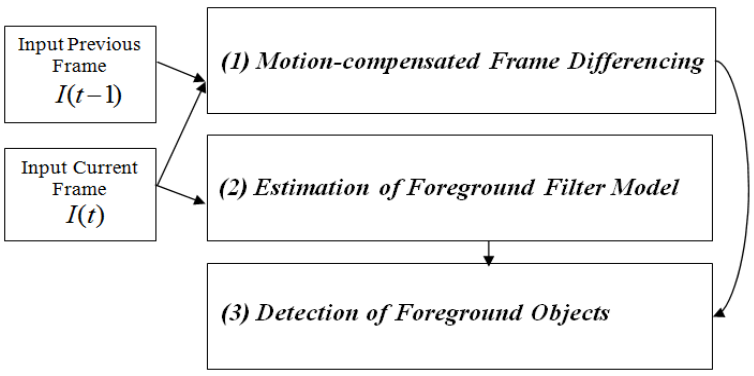


Figure 11.1: Diagram of the foreground object extraction method with the 3 main steps of the algorithm and their inputs.

DOI:10.15774/PPKE.ITK.2011.005

11.2 Motion-compensated Frame Differencing

In our case of a non static background, to know what parts of the picture are changing because of the camera motion and what parts are changing independently, the camera motion has to be estimated. To correctly align video frames we have used a Hierarchical Block-Matching (***HBM***) algorithm [10, 144]. It allows estimation of strong motion and has proven to be the best motion estimation approach in video coding applications, and it is also robust to local motion blur. The principle of HBM consists of dividing the current video frame $I(t)$ into a set of blocks. Then for each block, its best match is searched in frame I_{ref} by minimizing a sum of absolute difference criterion which is a function of a frame difference: $\Delta I(t) = |I(t) - \tilde{I}(t-1)|$. The difference of block center positions $\vec{d} = ((x_t - x_{t-1}), (y_t - y_{t-1}))$ is called a displacement vector. We refer the reader to [10, 144] for details of HBM, which allows estimating large displacements, up to 30 pixels in our case (see Fig. 11.2). If the HBM-based motion compensation

Figure 11.2: Three consecutive frames from a wearable outdoor video with strong motion

would work perfectly, than it would compensate not only the camera motion, but the motion of the foreground objects as well. Our goal is to model and compensate the motion of the camera only so that the foreground objects will still remain on the frames as camera independent motions. Therefore the motion vectors of the HBM are used as initial measures for a robust motion estimator [11] allowing for the rejection of outliers and obtaining a 6-parameter affine camera motion model:

$$\vec{d}(c_x, c_y) = \begin{pmatrix} a_1 \\ a_4 \end{pmatrix} + \begin{pmatrix} a_2 & a_3 \\ a_5 & a_6 \end{pmatrix} \begin{pmatrix} c_x \\ c_y \end{pmatrix} \tag{11.1}$$

DOI:10.15774/PPKE.ITK.2011.005

where $\vec{d}(c_x, c_y)$ is the displacement vector of the pixel block of center (c_x, c_y) and $a_1, .., a_6$ are the parameters of the camera motion model.

11.2.1 Creation of the Modified Error Image

After camera motion compensation our goal is to separate the moving objects from the background containing noise. Here we resort to the family of methods which model the probability density function of the background pixels and use it in the decision making. The approach we propose will neither use the original frame entirely, nor the simple frame difference. We propose a new measurement scheme building a signal we call "Modified Error Image" or MEI. After estimating and compensating the camera motion, the two consecutive images are aligned in the same coordinate system so that a frame difference can be calculated.

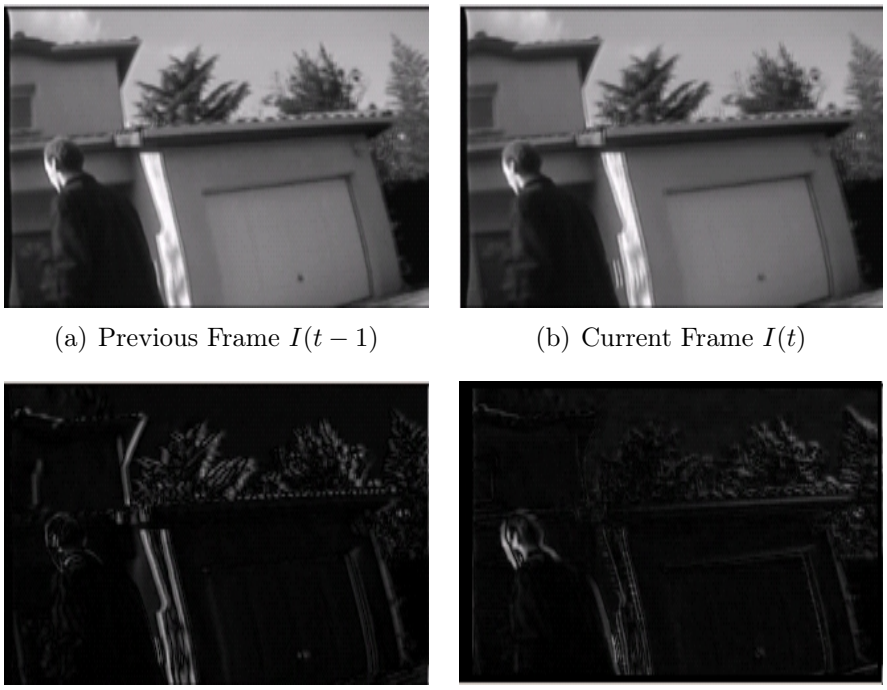
Let $I(t-1), I(t)$ be two consecutive frames. With (11.1) we transform $I(t-1)$ according to the camera motion between $I(t-1)$ and $I(t)$. We use this motion-compensated image, $\tilde{I}(t-1) = I((t-1), (x + \Delta x, y + \Delta y))$ to calculate an error image $E(t)$, which shows the pixels moving independently from the camera:

$$E(t) = \left| \tilde{I}(t-1) - I(t) \right| \tag{11.2}$$

Fig. 11.3 shows how the motion compensation enhances the result of frame differencing.

In the case of ideal camera motion compensation and in the absence of noise, the pixels with non-zero motion magnitude would be those that are moving in the real world. In reality, due to changes in the perspective, quantization error and motion blur, the highly contrasted contours in the scene will never be perfectly compensated. Thus the resulting error image will contain not only the pixels of a moving object, but false positive pixels too, making the direct detection of moving objects pixels impossible. Hence we propose to create a new, modified error image, on which the differentiation between static artifacts and moving objects could be done along time. The differentiation of static pixels and moving ones based only on gray level values is limited and we do not have any *a priori* information on the objects, hence for better discrimination, we would like to fully exploit the available color information. We propose to use the color information

DOI:10.15774/PPKE.ITK.2011.005



(a) Previous Frame $I(t-1)$ (b) Current Frame $I(t)$

(c) Difference Frame (d) Difference with motion compensation

Figure 11.3: The effect of motion compensation on frame differencing.

of the original image, thus the MEI at time t, will contain the color information of the original frame on those (x, y) pixels, where, the value of the error image $E(x, y, t)$ is significant. More formally, the modified error image, E^m is built as follows:

$$E^m(x, y, t) = \begin{cases} I(x, y, t), \text{ if } E(x, y, t) > th_E \\ 0 \text{ otherwise} \end{cases} \tag{11.3}$$

where I is a 3 channel frame taken by the camera at time t, E is the gray scale motion compensation error at time t. The threshold th_E helps filtering static, but non-zero value pixels from the error image. We experimentally fixed it to $th_E = 10$. Hence, the modified error image contains color information of the original frame, which will be used at the decision making step. Fig. 11.4 shows an example of the MEI. Contrary to approaches which use the whole original color frames for moving foreground detection, the MEI drastically saves computational

DOI:10.15774/PPKE.ITK.2011.005

(a) Original Frame (b) Error Image (c) Modified Error Image

Figure 11.4: An example of the Modified Error Image and its two sources: the original frame and the standard error image.

workload: only pixels of original frames, where the motion compensation error is strong, will be considered. According to our experiments, in average only 17% of the pixels of the original frame have to be processed. Furthermore, with motion compensated frame differencing and MEI building, we significantly eliminate complex background motions, thus reducing the overall complexity of object detection. In contrast to [13], where for each pixel a PDF-based decision is made on its label foreground/background, in our approach the PDF-based decision is done for false foreground removal only, as it is explained in the following section.

11.3 Estimation of Foreground Filter Model

The objective in this phase is to estimate a PDF of the color distribution associated to the background for each separate pixel on the MEI. This estimation is based on a short-term image history, in order to consolidate observations over several frames, after motion compensation. The assumption is that a pixel corresponding to a moving foreground object will have varying colors over such a time interval, whereas a pixel belonging to the background will have more stationary colors. We assume that large homogeneous foreground areas (e.g. car, bus) do not appear in a home environment and we don't have to face the so called foreground aperture problem. Comparing the current pixel color to its corresponding background model therefore allows refining detection by taking into account more frames. In order to decrease the computational complexity, this estimation is done only for pixels that have been segmented as potential foreground

DOI:10.15774/PPKE.ITK.2011.005

pixels during the motion-compensated frame differencing phase. The rest of this section is devoted to present how to estimate such a meaningful PDF.

11.3.1 Measurement Matrix

In order to estimate the PDF for background pixel values, we build a *measurement matrix* M. This matrix contains the information of the original frames, in n consecutive time instances and it is continuously updated along the time. Because of the unpredictable camera motion, a short temporal window is used for gathering frames to M (For the presented experiments a 15 frame long time window was used). In this way we can ensure that the frames in M have large overlapping parts, and are less affected by motion compensation errors. Updating at time t means adding the information of the current frame at time t to the measurement matrix of the previous time instance, $t-1$.

$$\begin{aligned} M(x,y,t) &= \Theta_t M(x,y,t-1) \cup I(x,y,t) \\ M(x,y,1) &= I(x,y,1) \end{aligned} \tag{11.4}$$

where the operator $\cup$ means adding new frame of measurements, while the oldest frame is being removed. Thus the number of frames in the matrix remains always the same. The operator Θ_t stands for the affine transformation with the estimated parameters of camera motion between time instance $t-1$ and t, (11.1). Applying this transformation we compensate all frames in the matrix to the reference frame, the current one.

11.3.2 Kernel Density Estimation

The measurements, stored in M, are used to estimate the probability that a new measurement belongs to the background. Considering each $f(x,y)$ pixel in the MEI as a realization of a random variable vector $X \in \mathbb{R}^3$, it is logical to suppose that if the pixel belongs to the background, then the realizations of X inside some temporal interval will follow the same PDF. As we stated in sub-chapter 10.4.1 both parametric and non-parametric PDF estimations are possible. Parametric estimation, such as for GMM model [15, 106], requires a large number of data for a reliable estimation of PDF parameters. As stated in [145], kernel density

DOI:10.15774/PPKE.ITK.2011.005

estimation [12] has become popular due to relaxed requirements on the quantity of data, hence in our work, taking into account the application scenario, we choose KDE.

The aim of this estimation method is to extrapolate the measured data into a regular density function. For the extrapolation, kernel functions, placed at each measurement point, are used with a smoothing parameter (See Section 10.4.1). We have considered using marginal or joint probability density estimation with diagonal bandwidth matrix (see Chapter 12 for test results):

$$H = \begin{bmatrix} \sigma^2[c1] & 0 & 0 \\ 0 & \sigma^2[c2] & 0 \\ 0 & 0 & \sigma^2[c3] \end{bmatrix} \tag{11.5}$$

where each σ^2 represents the bandwidth for a color channel.

The selection of the kernel function and the bandwidth parameter are obviously very important, since they both have a strong influence on the accuracy and the smoothness of the PDF estimate.

We use a sample point approach for bandwidth estimation, which is better suited to low sample cases than fixed bandwidth or balloon estimator [114, 146, 147]. It is defined as follows:

$$f_s(v) = \frac{1}{n}\sum_{i=1}^{n} K_{H(v_i)}(v - v_i) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{\|H(v_i)\|^{\frac{1}{2}}} K\left(H^{-1}(v_i)(v - v_i)\right) \tag{11.6}$$

where n is the number of measurements, and $H(v_i)$ is the sample-point bandwidth matrix associated to the ith sample point v_i.

In this work we were specifically interested in the case of low amount of samples. Furthermore, the "physics" of our problem implies some more constraints. Most of our samples are collected in the areas of highly contrasted borders, where the aliasing effect and the parallax could bring a strong error and hence a significant variation of values.

A common choice for the bandwidth calculation consists of using the distance of the sample point v_i from its kth nearest neighbor. However in our case the number of sample points is low and this kind of calculation might give false result,

DOI:10.15774/PPKE.ITK.2011.005

as pointed out in [148].

Choosing a high value for K would mean that the distance from the kth sample point to the point v_i can be high even if there are (at maximum $k-1$) other sample points near to v_i. In this case the information that there are other points near is completely lost. Choosing a low value for k, would mean that the distance might be low even if there are not many other samples near to the point, which makes the calculation sensitive to noise. Choosing a right k in the case of few measurements is not always possible.

To handle this problem we use the distance from all the k nearest neighbors, instead of using the distance from the kth alone. The $\sigma_i[c]$ parameter is calculated as the variance of the k nearest neighbors around the measurement v_i:

$$\sigma_i^2[c] = \frac{1}{k}\sum_{j=1}^{k}\left(v_i[c] - v_j[c]\right)^2 \tag{11.7}$$

where v_j is the jth nearest value to v_i in the measurement matrix c is the index of color channel. For the estimate to converge to the true unknown PDF, the following should be satisfied: $k(n)/n \to 0$ when $n \to \infty$. We use $k = \sqrt{n}$, where n is the number of available measurements.

It is generally accepted that the choice of the bandwidth is more important than the choice of the kernel function [126], although when the number of sample points is limited, the kernel function might have higher influence on the estimation. Several kernel functions (e.g.: Epanechnikov, Gaussian, uniform, triangle, quadratic, etc.) were tested and the Gaussian function proved to be the most suitable for our task (see Chapter 12). Choosing the Gaussian as kernel function the density estimator will be:

$$f_{x,y}(v) = \frac{1}{n(2\pi)^{d/2}}\sum_{i=1}^{n}\frac{1}{\|H(v_i)\|^{\frac{1}{2}}}e^{-\frac{1}{2}\left((v-v_i)^T H^{-1}(v_i)(v-v_i)\right)} \tag{11.8}$$

11.3.3 Spatial-Temporal Selection of the Measurement Points

To cope with the lack of data, due to limited temporal history, we propose a new concept of a spatial-temporal PDF which will be explained in this section. We call it spatial-temporal according to the choice of sample points: we use both

DOI:10.15774/PPKE.ITK.2011.005

spatial neighborhood and temporal history.

In [15] and [13] the sample points at given (x, y) coordinates are the n previous measurements taken at the same (x, y) position: $(v(x, y, 1), v(x, y, 2), ..., v(x, y, n))$. However, when the camera is moving the case is different: even after motion compensation the real background scene position that corresponds to the (x, y) pixel in one frame, might move a little, due to minor errors of camera motion compensation or quantization. Assuming that this error is random, the use of a small (x, y) centered patch can solve the problem.

We have tested two different methods for gathering measurements from it. The straightforward idea is to use all points from the patch. In this case some noise might be added to the data but we can increase the number of measurements significantly. The other idea is to use that pixel from the patch which might correspond to the pixel in question. To find it we look for the closest pixel in color space. This way the data contains less noise but the number of measurements remains low. We have made experiments with both methods and based on the results we have chosen the first approach (see Chapter 12 for more detailed explanation).

Based on the values of the measurement matrix, probability density functions are built for each non zero (x, y) pixel of the current modified error image using (11.8), where vi is the previously measured value of the (x, y) point, obtained from the M matrix through spatio-temporal selection, n is the number of measurements, $H_i = H(v_i)$ is the bandwidth matrix (see (10.13) and (11.7)).

We have to note that n is an effective maximal number of the non-zero measures available for PDF building. In practice the number of available measurements can change for each pixel. In the case when there is no measurement for a pixel we do not apply the kernel-based filter, thus these pixels will remain on the Filtered Error Image as foreground.

11.4 Classification of Foreground/Background Pixels

Once the PDF has been built for each pixel in the current MEI, we can proceed to the detection of moving foreground objects. Here the pixels will be first classified as belonging to the foreground or background on the basis of the PDFs charac-

DOI:10.15774/PPKE.ITK.2011.005

teristics. Then the detected pixels will be grouped into clusters (moving objects) on the basis of their motion, color and spatial coordinates in the image plane.

The $f_{x,y}(v)$ function is a PDF that shows how likely the pixel (x, y) takes a value v. Based on this likelihood we want to divide the domain R of all possible v values into two parts: R_1 and R_2. R_1 is associated to the background colors and R_2 to the foreground colors. If we measure a value which is in R_1, it will be classified as background; otherwise it will be classified as foreground. The union of R_1 and R_2 has to be equal to R which is the whole domain. When classifying a measured value into background or foreground two kinds of mistakes can be made: classify a background point as foreground and classify a foreground point as background. Let p_1 be the background PDF and p_2 be the foreground PDF. Then the probabilities of misclassification are:

$$P(2|1, R) = \int_{R_2} p_1(v)\mathrm{d}v \tag{11.9}$$

$$P(1|2, R) = \int_{R_1} p_2(v)\mathrm{d}v \tag{11.10}$$

In our case $P(2|1, R)$ is the probability of false detection of an object pixel and $P(1|2, R)$ is a missed detection of an object pixel. If both PDFs would be known, we could find an optimal division of R that minimizes the two kinds of error: $P(2|1, R) + P(1|2, R)$. However the PDF of the foreground is not known in our case, hence we propose a threshold-based decision scheme using only the background PDF.

11.4.1 Adaptive Threshold Calculation

Our goal now is to keep $P(2|1, R)$ small, while ensuring that the territory of the background remains as small as possible. We define $R1$ and $R2$ as follows:

$$R_1 = \{v \in R | f_{x,y}(v) \geq T_{x,y}\} \tag{11.11}$$

$$R_2 = \{v \in R | f_{x,y}(v) < T_{x,y}\} \tag{11.12}$$

DOI:10.15774/PPKE.ITK.2011.005

where $f_{x,y}(v)$ is the PDF of the background and $T_{x,y}$ is a threshold. The higher the $T_{x,y}$, the smaller the R_1, and the bigger the $P(2|1,R)$. By calculating the integral of $f_{x,y}$ over R_2, $P(2|1,R)$ can be controlled and kept under a predefined α (misclassification probability of the background):

$$\int_{R_2} f(v)\mathrm{d}v = P(2|1,R) < \alpha \tag{11.13}$$

Determining T based on this integral requires too much computational power. To avoid this, we use a simple heuristic. First let us introduce a heuristic property, the Efficiency Measure (***EM***) for a PDF, which shows how efficiently the PDF can be covered by a closed sub-domain:

$$EM = \frac{\int_{|r|} f(x)\mathrm{d}x}{|r|} \tag{11.14}$$

where $|r|$ is the measure of the sub-domain. Our heuristic claims that the average height of the function in the sample points v_i (where the Gaussians are centered) is proportional to EM (and thus inversely proportional to the territory needed to cover it).

$$\frac{\sum_{i=1}^{n} f(x_i)}{n} \propto EM \tag{11.15}$$

If we choose the threshold to be proportional to that average, than it will be higher (and $R1$ will be smaller) if the EM is high, and lower ($R1$ will be bigger) when the EM is small. It means that the threshold will be adapted to the shape of the PDF. After this consideration we have chosen the $T_{x,y}$ threshold as follows:

$$T_{x,y} = \lambda \frac{\sum_{i=1}^{n} f_{x,y}(v_i)}{n} \tag{11.16}$$

where λ is a constant, n is the number of the available measurements and $v_i \in M(x,y)$.

11.4.2 Decision-Making Rule

Once the adaptive threshold has been defined, the classification of pixels into foreground/background is straightforward. We perform it on the modified error

DOI:10.15774/PPKE.ITK.2011.005

image $E_m(x, y, t)$: if the new value v has a high probability at the given (x, y) position then we consider it as part of the background and eliminate it from the error image:

$$E_f^m(x, y, t) = \begin{cases} E^m(x, y, t) \text{ if } f_{x,y}(v) \leq T \\ 0 \text{ otherwise} \end{cases} \tag{11.17}$$

The resulting $E_f^m(x, y, t)$ is the filtered error image, which contains the points of the foreground.

The typical results of this classification are depicted in Fig. 11.5. Due to the noise and errors in motion estimation, this detection result still contains some noise, such as isolated pixels. Furthermore, the foreground pixels are spread over the frame while in the context of our problem, the moving objects should represent compact areas in the image plane. We can reasonably suppose that with the high frame rate we have, the pixel displacements of objects are similar. Hence we propose to cluster detected foreground points in a mixed feature space, supposing that an object will be represented by one single cluster or by a set of clusters close to each other in the image plane.

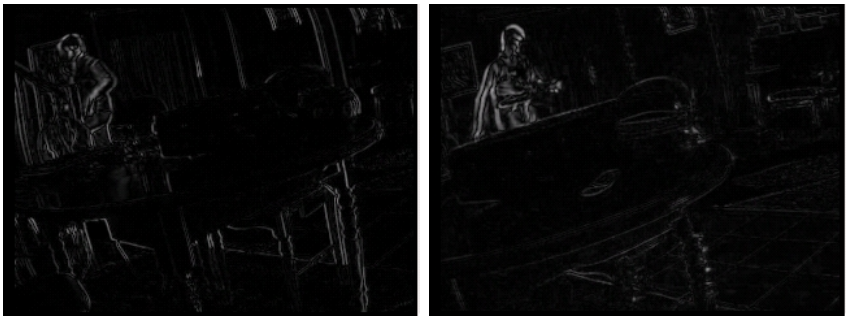
11.5 Clustering of Foreground Points with DBSCAN

To find moving object silhouettes and eliminate the remaining noise, we used a clustering algorithm, called DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [14] in a mixed feature space R^l. In this space with $l = 7$ dimensions, each foreground point is described with a feature vector $X = (x, y, C1, C2, C3, dx, dy)^T$ which contains the x, y coordinates, the color coordinates $C1, C2, C3$ in normalized RGB space and the coordinates of a displacement vector dx, dy expressing pixel motion. Intuitively this means that the points that are close to each other, moving together and have similar color will be put in the same cluster. DBSCAN is a density-based clustering algorithm that can separate arbitrary shaped clusters. The main advantages of DBSCAN are: it does not require knowing *a priori* the number of clusters, does not have a bias towards a particular cluster shape or size and it is resistant to noise [14], [149]. On the other hand, it does not work well on high dimensional data or a dataset with varying density. For all these reasons presence of detection noise, low dimen-

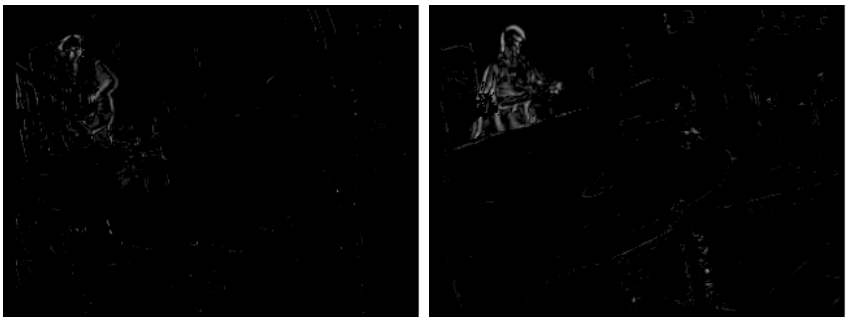
DOI:10.15774/PPKE.ITK.2011.005

sionality of the feature space, arbitrary shape of presumed clusters we found it to be convenient for our problem. The results of the clustering can be seen in Fig. 11.5(c). It shows the clusters obtained with DBSCAN from a raw foreground detection results (Fig. 11.5(b)). One can see that a lot of detection has been filtered. The bounding boxes of the clusters are superimposed on the original frame in Fig. 11.5(d).

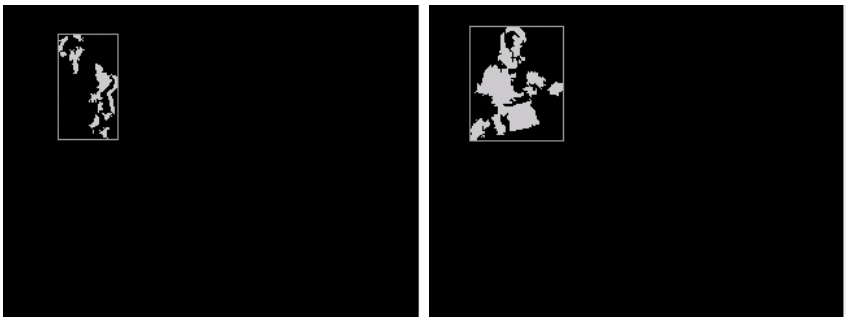
DOI:10.15774/PPKE.ITK.2011.005



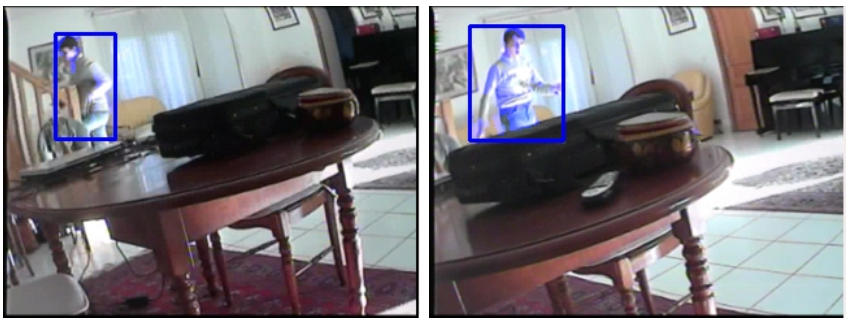
(a) Error Image



(b) Filtered Error Image



(c) Objects Detected by DBSCAN



(d) Detected Foreground Objects

Figure 11.5: The main steps of the foreground object detection.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 12

Experiments

The results to be presented were obtained on healthy volunteers and real patients. In order to keep the conditions of the experiment ergonomic for the observed subjects, the videos show their standard everyday conditions. They usually stay alone at home. This is why the sequences containing moving objects (persons, animals) are very rare. In the corpus of duration of 9 hours 17 minutes moving objects (persons in our case) only occur occasionally for short periods of a few seconds. Hence to construct the ground truth for the tests of our method, we mainly used these short sequences. Some key frames and the corresponding results are presented in Fig. 12.10. A sample of the dataset and the belonging ground truth on healthy volunteers is publicly available at:

http://www.labri.fr/projet/AIV/projets/peps/

In order to assess the false detection rate of our method experiments on sequences without moving objects will also be presented.

12.1 Evaluation Metrics

To evaluate the performance of the method we used F-score [150]:

$$F = \frac{2}{\frac{1}{Re} + \frac{1}{Pr}} \tag{12.1}$$

where Re is the detection rate (recall) and Pr is the positive detection rate (precision). The Recall and Precision were measured using two kinds of Ground Truth (GT) data. For comparison with a base-line method [90, 91] we used handmade rectangular shaped GT. Every pixel inside the GT area was considered as foreground and every pixel outside the GT rectangular as background. During the search for the best parameters we used a modification of the above described GT. The modifications will be explained in details later.

DOI:10.15774/PPKE.ITK.2011.005

12.2 Comparison with a Base-line Method: Gaussian Mixture Model

As a base-line method we used a variety of Stauffer and Grimson's GMM method [92]. This alternative method is based on [90,91], with additional selection of the number of the Gaussian components: [151]. Both GMM and Kernel-based model were tested on motion compensated images. Here GMM is used as a background model: the pixels that do not fit to the model will be foreground pixels. The maximum number of Gaussians in the method [91, 92] was fixed as $K = 4$. The initial bandwidth for a new mode was chosen $\sigma = 11$ and the complexity reduction prior constant was $CT = 0.2$. The method was used without shadow detection. The results of the detection as a function of the learning parameter α, which tunes the update of Gaussians, are given in 12.1(a). For detailed description of the method see [91, 92]. The best result obtained for GMM in terms of F-Score was at $\alpha = 0.75$ where $F = 0.156$.

In order to test the effectiveness of the Modified Error Image 11.3 as pre-filter, we have taken the MEI as initial foreground mask and confirmed it by detected pixels with GMM method applied to motion compensated frames. This is the same concept as the Kernel-based density estimation that was used in our method (see Section 11.4.2). The received F-score was higher with this pre-filtering concept: $F = 0.183$. Hence the MEI not only saves a lot of computational time, but increases the effectiveness of the foreground detection.

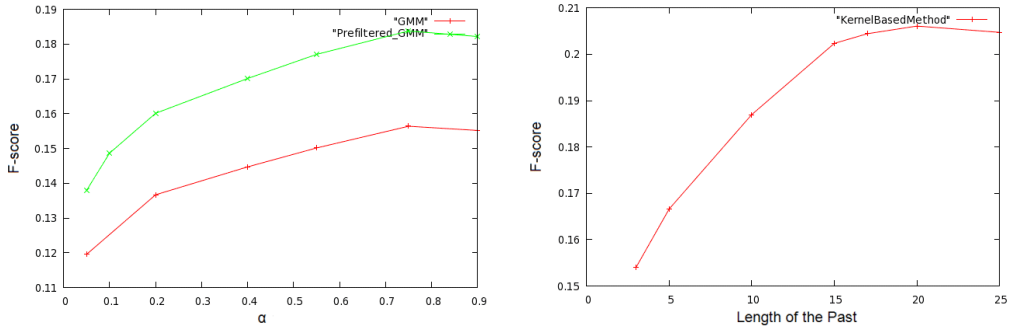
Exchanging the GMM with the proposed Kernel-based estimation with Gaussian kernel shows further improvement: $F = 0.206$. Fig. 12.1 shows the results of

	GMM	**GMM filter**	**Kernel-based filter**
Peak F-score	0.156	0.183	0.206
Precision	0.0938	0.114	0.152
Recall	0.469	0.458	0.316

Table 12.1: Peak F-scores for the base-line and the Kernel-based method

GMM, GMM filter and Kernel-based filter methods as a function of dependency on the previous frames. The peak F-scores are summarized in Table 12.1. The best result in the case of Kernel-based filter was obtained at $n = 20$. Since the

DOI:10.15774/PPKE.ITK.2011.005



(a) The GMM with and without pre-filtering as a function of learning parameter (α)

(b) Kernel-based method as a function of the number of previous frames considered for building the PDF

Figure 12.1: Results obtained with Gaussian Mixture Model and the proposed Kernel-based filtering.

F-score does not change much between $n = 15$ and $n = 20$, to save computational power we decided to use $n = 15$. The F-score at $n = 15$ is 0.202.

12.3 Step-by-Step Validation of the Kernel-based Filtering Method

As Table 12.1. shows, Kernel-based method gives better results in the same circumstances; hence it was chosen over GMM for calculating PDF for each candidate foreground point, based on previous measurements. The question is what parameter set (color space, measurement point selection, patch size, etc.) is the most suitable for our task, where a special difficulty is presented by the strongly limited number of measurements

In the following we will test Kernel-based filters on the MEI. To better evaluate the filters, we introduce a new Ground Truth ($\boldsymbol{GT}$). So far we used a handmade GT, where the true foreground was marked with rectangular shaped areas. From now on we will restrict these true foreground points to those pixels that are non-zeros on the MEI: the true foreground can be created with a logical AND between the handmade GT and the MEI.

This modification makes sense since our initial data for Kernel-based filtering

DOI:10.15774/PPKE.ITK.2011.005

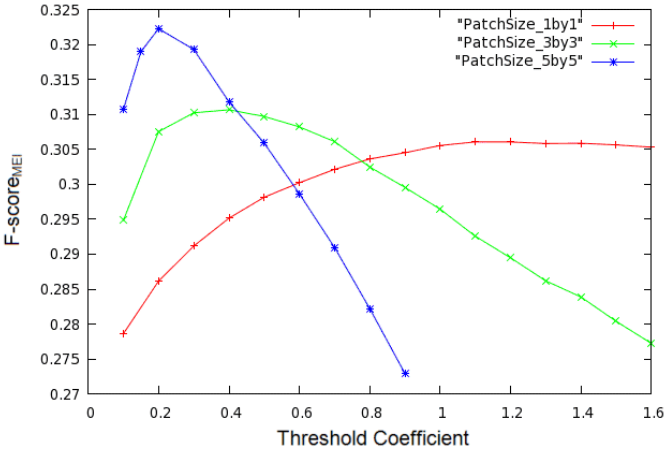


Figure 12.2: Results obtained with different patch sizes: 1x1, 3x3, 5x5

is the MEI, which means that only the non-zero pixels of the MEI has the chance to be on the final foreground mask. Note that, since the GT is different, the following results are not directly comparable with the results of the previous subsection. To be unambiguous the F-score values, calculated on this modified GT will be named as F-score$_{\text{MEI}}$.

12.3.1 Patch Size

If only few consecutive frames are available, it is natural to use measurements from the surrounding area of a pixel. It will not only raise the number of measurements but might help dealing with smaller motion compensation errors (see Section 11.3.3). We compared the F-score$_{\text{MEI}}$ of kernel methods in the case of 1x1, 3x3 and 5x5 sized patches as a function of probability threshold coefficient, (11.16). See Fig. 12.2. While in the case of a 1x1 patch size the F-score$_{\text{MEI}}$ is more stable, with a larger patch size it has higher peak and it drops very quickly. This can be explained by the following: the threshold values are calculated as a function of average kernel heights. In the case of a larger patch the height of the kernels will be higher, since the measurement values are closer to each other and this results small sigma values. If the threshold is higher, the changes of the coefficient have greater impact on the result. The test results confirmed that larger patch size is more suitable in our "wearable" case (see Table 12.2. for summary). For sake of

DOI:10.15774/PPKE.ITK.2011.005

Patch Size	1x1	3x3	5x5
Peak F-score$_{\mathrm{MEI}}$	0.306	0.310	0.322

Table 12.2: Peak F-scores$_{\mathrm{MEI}}$ obtained with different patch sizes

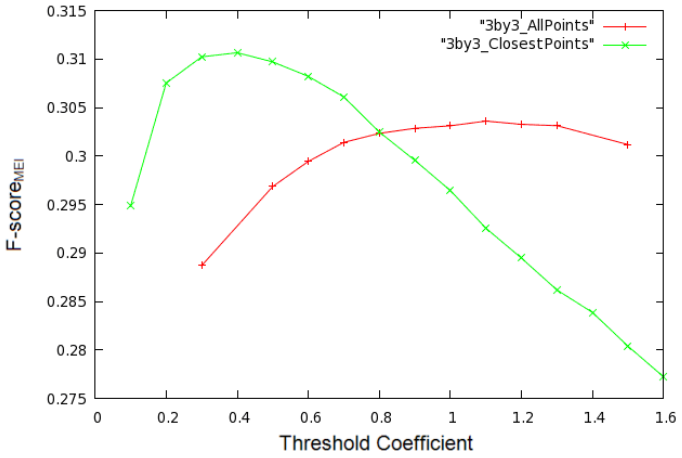


Figure 12.3: Results obtained with different point selection techniques, both with marginal distribution

computational complexity we used 3x3 sized patches.

12.3.2 Measurement Point Selection Techniques for Joint and Marginal Representation

If not only previous pixel values are used as measurements, but the measurement values are selected from a patch, then different methods can be used for selecting points from the patch. Here we compare two ways for measurement selection (see Section 11.3.3). The first is to use all values from the patch, the second is to use only the closest value in the color space. Fig. 12.3. shows the results obtained by different point selection techniques in the case of a 3x3 patch. The corresponding peak F-score$_{\mathrm{MEI}}$ values can be found in Table 12.3. We can see that selecting only the closest point from a patch gives better results than selecting all points from it. Selecting the closest value helps correcting small errors of motion compensation without adding too much noise to the estimation.

However, using only one point from a patch does not increase the number

DOI:10.15774/PPKE.ITK.2011.005

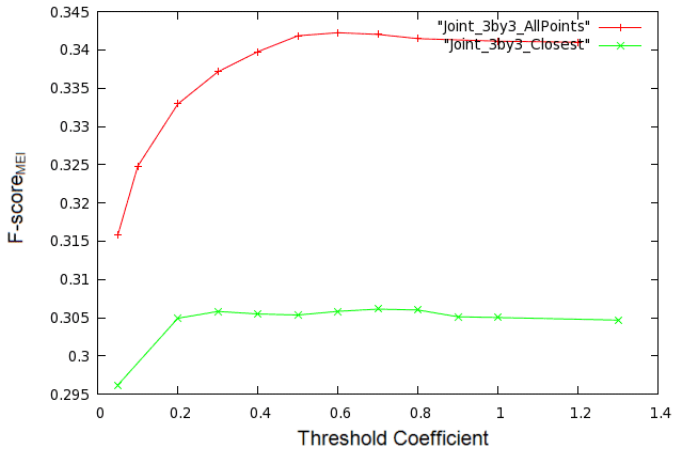


Figure 12.4: Results obtained with "all points" and "closest point" selection techniques, both using joint distribution

of measurements, just make the measurements more accurate. In the case of marginal PDF calculation it enhances the results, but if we use joint distribution PDF over the color space we can see that the number of measurements is not enough with respect to the number of dimensions. This explains why the all value selection method works better in the case of joint distribution, as can be seen on Fig. 12.4. Table 12.3 shows the best results obtained with joint and marginal distribution with optimal patch size and measurement selection method.

Distribution	Marginal					Joint		
Patch Size	1x1	3x3		5x5		1x1	3x3	
Point Selection	N/A	All	Closest	All	Closest	N/A	All	Closest
Peak F-score$_{\text{MEI}}$	0.305	0.303	0.310	N/A	0.322	0.304	0.342	0.306

Table 12.3: The best results obtained with joint and marginal distribution

12.3.3 Effect of the Choice of the Color Space

We also have examined the performance of the filter in different color spaces. Fig. 12.5. shows the measurements taken in RGB, normalized RGB, HSV, and YUV color spaces. We obtained the best results in normalized RGB color space. Although HSV and RGB color spaces are more stable, since the threshold coef-

DOI:10.15774/PPKE.ITK.2011.005

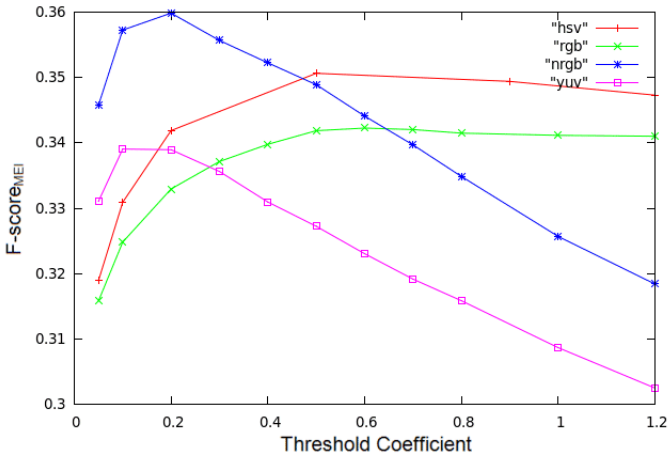


Figure 12.5: Results obtained in 4 different color spaces

ficient is a chosen constant (see (11.16)), it is more important that the highest value of the curves is better for nRGB color space.

12.3.4 Effect of the Choice of the Kernel Function

The method was tested with different Kernel functions: Gaussian, Quadratic, Tricube, Epanechnikov, Triangle and Uniform kernels. Fig. 12.6 shows the obtained results. We got the best result with Gaussian kernel function, which is the smoothest of the tested kernel functions and this property has high importance in the case of a small number of measurements.

12.3.5 Choice of the Kernel Width

Here we compare three methods for kernel width calculation: using a constant value for bandwidth or the distance from kth nearest neighbor or the average distance from the closest k nearest neighbors. Using fixed bandwidth gives significantly lower F-score$_{\text{MEI}}$ values than the other two: the peak F-score$_{\text{MEI}}$ is 0.302. The other two methods show similar results, however using the k-nearest neighbors gives slightly higher F-score$_{\text{MEI}}$ (see Fig. 12.7.)

DOI:10.15774/PPKE.ITK.2011.005

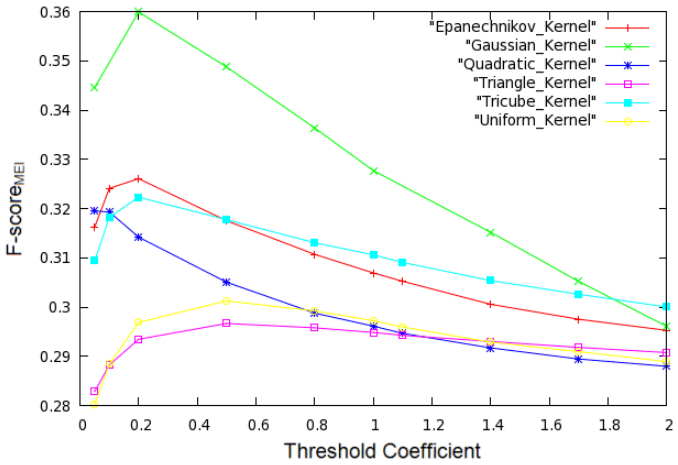


Figure 12.6: F-scores$_{\text{MEI}}$ with different Kernel functions as a function of threshold coefficient.

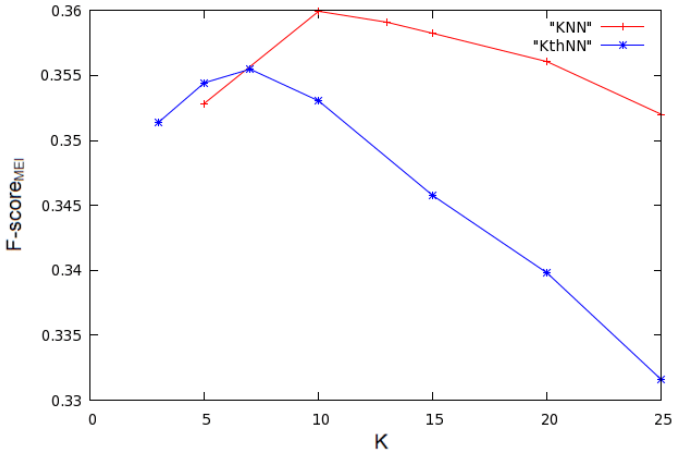


Figure 12.7: Comparison of kNN and kthNN bandwidth selection methods.

DOI:10.15774/PPKE.ITK.2011.005

Density Estimation Method	Probability Representation	Patch Size	Point Selection Method	Color Space	Kernel Function	Kernel Width
Gaussian Mixture Model	Marginal	1x1	**All**	RGB	**Gaussian**	Fix
				nRGB	Epanechnikov	
		3x3		HSV	Tricube	kth NN
Kernel-based Estimation	**Joint**		Closest		Uniform	
		5x5		YUV	Triangle	k **NN**
					Quadratic	

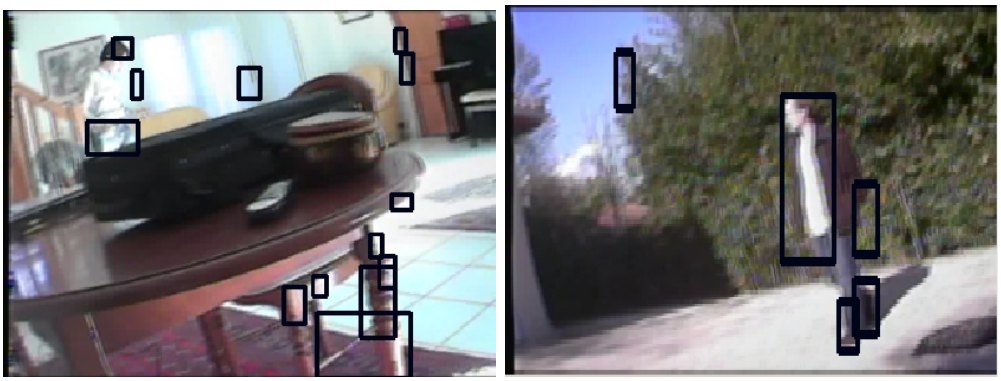
Table 12.4: Summary of the decisions at parameter selection. Our choices are highlighted in bold.

12.4 Overall Detection Performance of the Proposed Method

The proposed method was compared to a GMM-based foreground object detection described in [92] (implementation available at [151]). Table 12.5 shows the final results of the proposed method and the alternative method (Gaussian Mixture Model based method) on videos acquired with a standard button camera. The results we obtain are almost 3 times better on these complex sequences than those of the method [92]. Some example results of the compared methods are given in Fig. 12.8. In these experiments moving objects were shot by a standard camera and the persons were not very close to the device. The ground truth was made by hand for all the sequences. The precision and recall rates were calculated frame-by-frame based on the overlap between the pixels annotated as foreground in the ground truth and the estimated foreground image in the case of both methods. Both the ground truth and the estimated foreground are rectangular shaped. See the illustration on Fig. 12.9. Our proposed method performs better both in recall and precision metrics.

The main reason of the relatively low precision and recall values is the low quality of the frames, especially when they are corrupted by motion blur, which corrupts the results of the camera motion compensation. Many false detections are caused by static objects very close to the camera. These objects seem as if they were moving due to the change of the camera perspective.

DOI:10.15774/PPKE.ITK.2011.005



(a) GMM)

(b) Proposed Kernel-based method

Figure 12.8: Example images of foreground detection

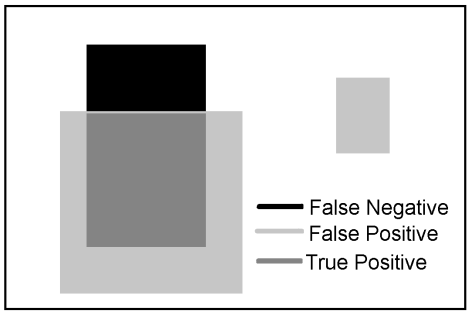
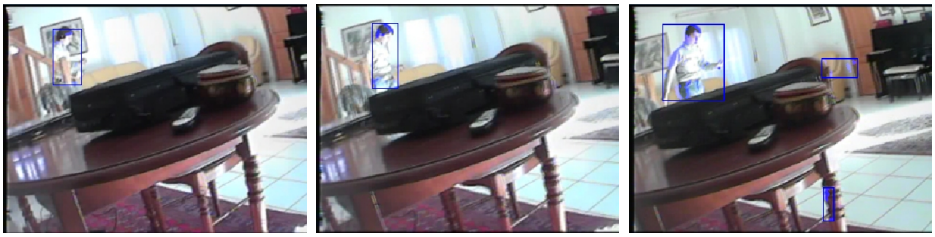
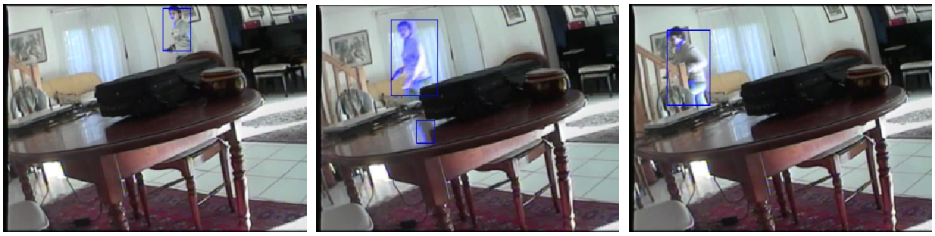


Figure 12.9: Illustration of the regions used for evaluation.

DOI:10.15774/PPKE.ITK.2011.005

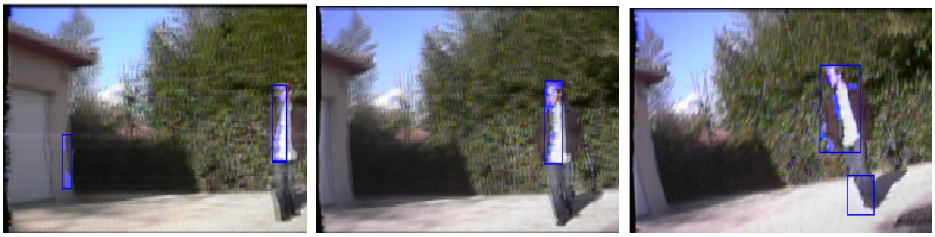


(a) François 1



(b) François 2

(c) Daniel 1



(d) Daniel 2

Figure 12.10: Example of pictures from the tested sequences.

DOI:10.15774/PPKE.ITK.2011.005

Precision/Recall/F-scorel			
Sequence Name	# Frames	GMM-based method	Proposed method
Francois 1 (Indoor)	60	0.050 / 0.332 / 0.087	0.365 / 0.906 / 0.52
Francois 2 (Indoor)	141	0.242 / 0.331 / 0.279	0.801 / 0.799 / 0.80
Daniel 1 (Outdoor)	90	0.267 / 0.428 / 0.329	0.624 / 0.574 / 0.598
Daniel 2 (Outdoor)	30	0.236 / 0.302 / 0.265	0.467 / 0.772 / 0.582

Table 12.5: Precision, recall and F-score rates for 4 different sequences for the proposed and a concurrent method.

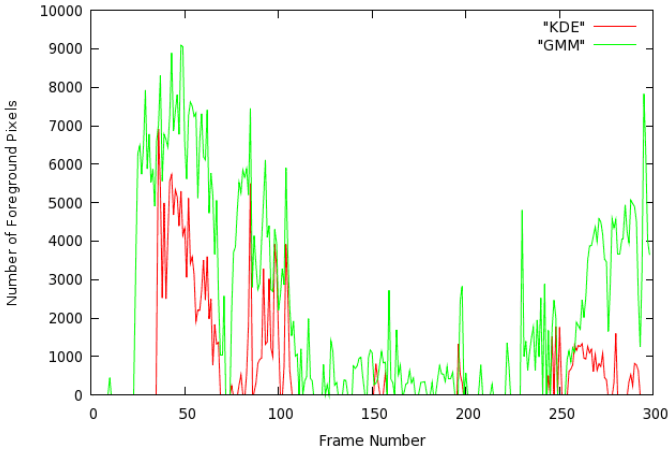


Figure 12.11: The number of false foreground pixels on an empty sequence.

12.5 Experiments on "Empty" Sequences

To assess false detection rate of our method we have also made experiments on "empty" sequence (see Fig. 12.11.) where no moving objects were available. Both methods give false positives, but our proposed method is more than 70% better in average (lower curve in Fig. 12.11.).

12.6 Time Performance

The algorithm was tested on Intel Pentium 4, 3.4 GHz CPU, 1 GB memory with Linux operating system. In the present state the run time of the algorithm is far from real-time (see Table 12.6.), since we use non optimized software implementation without hardware acceleration. This therefore requires off-line processing of the recorded data. Since the most time consuming steps are well parallelizable,

DOI:10.15774/PPKE.ITK.2011.005

Time Consumption				
Block-Matching	Motion Compensation	Kernel-based Foreground Filtering	Clustering	Overall
16.0209	0.1210	23.2958	0.0761	39.4138

Table 12.6: Time consumption of the main steps of the algorithm in seconds.

the use of GPGPU can be a perspective for real-time processing. As Table table: Bordeaux: Time Consumption. shows, the most time consuming steps are the Block-Matching and the Kernel-Based Foreground Filtering. The parallelization of the former is well studied in the literature ([152, 153]). For the latter let us examine its time consumption in detail. Table 12.7. shows the computational

Time consumption of Kernel-based Foreground Filtering of one patch				
One kernel (one component of eq.(11.7))	One PDF value eq.(11.7)	Bandwidth of a kernel	Threshold	Overall
0.0044	0.0314	0.8918	8.0261	8.9537

Table 12.7: Time consumption of the Kernel-based Foreground Filtering of one patch in milliseconds

time needed for processing an average patch. It can be seen that the calculation of one patch takes only a few milliseconds and as the patches are independent from each other a naive way of parallelization is to handle each one of them as an independent thread.

Another approach could be going down to pixel level. According to our measurements the most time demanding step in the processing of a patch is the calculation of the threshold, which is essentially the repeated calculation of (11.7). (11.7) is the sum of Gaussian kernel values that can be calculated independently. Hence a promising way of parallelization is the parallel processing of the components of (11.7). A drawback in this case is the increased number of memory accesses compared to the patch-based decomposition.

These are two ways of breaking down the problem into parallel threads. Other, more sophisticated ways of parallelization may also be studied, but this is not the subject of this dissertation.

DOI:10.15774/PPKE.ITK.2011.005

Chapter 13

Conclusion and Perspectives

In this part a novel method was proposed for moving foreground object extraction based on kernel density function estimation on sequences taken by a wearable camera with strong and unpredictable motion.

The camera motion was estimated and compensated with a block matching based global motion estimator, and motion-compensated frame differencing was applied for change detection. To enhance the result of the frame differencing a novel, kernel-based PDF foreground filter model was proposed to eliminate false detections. We followed the approach of Kn nearest neighbors with a small amount of measurements and proposed a novel scheme for the choice of the scale parameter of Gaussian kernels. To detect moving foreground pixels an adaptive thresholding scheme was proposed. On the remaining foreground points the DB-SCAN clustering algorithm was used to build foreground objects from the points. It allowed for elimination of isolated noisy detection results thus reducing the false detections.

Our work in separation of the foreground and the background is just the first step toward content based search of videos, which is one of the most intensively researched areas of multimedia and computer vision. It has a lot of potentials in security or medical surveillance and basically in all the applications where moving cameras are used.

DOI:10.15774/PPKE.ITK.2011.005

Part IV

Conclusions and Perspectives

DOI:10.15774/PPKE.ITK.2011.005

In this dissertation low- and mid-level image processing algorithms were presented that could help the processing of video recordings taken by a wearable camera, where the quality of the signal is lowered by motion blur and noise.

In Part I. an automatic procedure have been described for estimating the stopping condition of non-regularized iterative deconvolution methods based on ADE independence measure, calculating the orthogonality of the estimated signal and its gradient at a given iteration. This method outperforms the generally applied *ad-hoc* stopping conditions and it may help to better reconstruct motion blurred frames.

Part II. presents an image decomposition method that splits the image into cartoon and texture (or noise) parts using anisotropic diffusion with orthogonality based parameter estimation and stopping condition, utilizing the theory that the cartoon and the texture components of an image are independent of each other. Based on our experiments the presented method outperforms the stat-of-the-art algorithms.

A method for moving foreground object extraction in wearable camera recordings has been introduced in Part III. Camera motion compensated frame differencing is applied and enhanced with kernel density estimation of the PDF of background pixels. The algorithm was thoroughly tested and compared to a generally known baseline method with good results.

Our work in separation of the foreground and the background is just the first step toward content based search of videos, which is one of the most intensively researched areas of multimedia and computer vision. The decomposition of an image into cartoon and texture components could be useful in motion estimation to eliminate the effect of noise that often causes false results. Deconvolution methods are widely used in image processing where defocusing is an issue: from microscopy to astronomy. Also, it could be used as preprocessing of videos taken by moving cameras, where motion blur corrupts the frames.

DOI:10.15774/PPKE.ITK.2011.005

Bibliography

[1] L. Kovács and T. Szirányi, "Focus area extraction by blind deconvolution for defining regions of interest," *IEEE Tr. PAMI*, vol. 29, pp. 1080–1085, 2007.

[2] W. Richardson, "Bayesian-based iterative method of image restoration," *JOSA*, vol. 62, pp. 55–59, 1972.

[3] L. Lucy, "An iterative technique for rectification of observed distributions," *The Astronomical Journal*, vol. 79, pp. 745–765, 1974.

[4] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 629–639, 1990.

[5] A. Buades, T. Le, J.-M. Morel, and L. Vese, "Fast cartoon + texture image filters," *IEEE Transactions on Image Processing*, vol. 19, no. 8, pp. 1978 – 1986, 2010.

[6] W. Yin, D. Goldfarb, and S. Osher, "Image cartoon-texture decomposition and feature selection using the total variation regularized L1 functional," in *Variational, Geometric, and Level Set Methods in Computer Vision*, pp. 73–84, Springer, 2005.

[7] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, pp. 259–268, 1992.

[8] F. Zhang, X. Ye, and W. Liu, "Image decomposition and texture segmentation via sparse representation," *Signal Processing Letters, IEEE*, vol. 15, pp. 641 –644, 2008.

[9] R. Shahidi and C. Moloney, "Decorrelating the structure and texture components of a variational decomposition model," *IEEE Transactions on Image Processing*, vol. 18, no. 2, pp. 299 –309, 2009.

[10] M. Bierling, "Displacement estimation by hierarchical block matching," pp. 942–951, 1988.

DOI:10.15774/PPKE.ITK.2011.005

[11] M. Durik and J. Benois-Pineau, "Robust motion characterization for video indexing based on mpeg2 opticalflow," *In Proc. of the International Workshop on Content-Based Multimedia Indexing*, pp. 57–64, 2001.

[12] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[13] A. Mittal and N. Paragios, "Motion-based background subtraction using adaptive kernel density estimation," vol. 2, pp. 302 –309, 2004.

[14] M. Ester, H. peter Kriegel, J. S, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," pp. 226–231, AAAI Press, 1996.

[15] C. Stauffer and W. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 747 –757, 2000.

[16] S. Hodges, L. Williams, E. Berry, S. Izadi, J. Srinivasan, A. Butler, G. Smyth, N. Kapur, and K. Wood, "Sensecam: a retrospective memory aid," *International Conference on Ubiquitous Computing, LNCS 4206*, pp. 177–193, 2006.

[17] N. Sprljan, M. Mrak, and E. Izquierdo, "Image compression using a cartoon-texture decomposition technique," *Proc. Int. Work. on Image Analysis for Multimedia Interactive Services (WIAMIS)*, p. 91, 2004.

[18] M. Kunt, A. Ikonomopoulos, and M. Kocher, "Second-generation image-coding techniques," *Proceedings of the IEEE*, vol. 73, no. 4, pp. 549 – 574, 1985.

[19] D. Barba and J.-F. Bertrand, "Optimization of a monochrome picture coding scheme based on a two-component model," in *9th International Conference on Pattern Recognition, 1988.*, pp. 618 –622 vol.1, nov 1988.

[20] S. C. L. Zou, H. Zhou and C. He, "Dual range deringing for non-blind image deconvolution," *International Conference on Image Processing*, pp. 1701–1704, 2010.

DOI:10.15774/PPKE.ITK.2011.005

[21] A. K. Jain, *Fundamentals of digital image processing*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1989.

[22] J. G. McNally, T. Karpova, J. Cooper, and J. A. Conchello, "Three-dimensional imaging by deconvolution microscopy," *Methods*, vol. 19, no. 3, pp. 373 – 385, 1999.

[23] A. N. Tikhonov and V. Y. Arsenin, *Solutions of ill-posed problems*. Scripta series in mathematics, Washington: Winston, 1977.

[24] A. Erhardt, G. Zinser, D. Komitowski, and J. Bille, "Reconstructing 3-d light-microscopic images by digital image processing," *Applied Optics*, vol. 24, pp. 194–200, 1985.

[25] T. Tommasi, A. Diaspro, and B. Bianco, "3-d reconstruction in optical microscopy by a frequency-domain approach," *Signal Processing*, vol. 32, no. 3, pp. 357 – 366, 1993.

[26] W. A. Carrington, R. M. Lynch, E. D. W. Moore, G. Isenberg, K. E. Fogarty, and F. S. Fay, "Superresolution Three-Dimensional Images of Fluorescence in Cells with Minimal Light Exposure," *Science*, vol. 268, pp. 1483–1487, 1995.

[27] P. A. Jansson, R. H. Hunt, and E. K. Plyler, "Resolution enhancement of spectra," *J. Opt. Soc. Am.*, vol. 60, no. 5, pp. 596–599, 1970.

[28] D. Agard, "Optical sectioning microscopy: Cellular architecture in three dimensions," *Ann. Rev. Biophys. Bioeng*, vol. 13, pp. 191–219, 1984.

[29] D. A. Agard, Y. Hiraoka, P. Shaw, and J. W. Sedat, "Fluorescence microscopy in three-dimensions," *Methods in Cell Biology*, vol. 30, pp. 353–377, 1989.

[30] P. Krämer, J. Benois-Pineau, and J.-P. Domenger, "Local object-based super-resolution mosaicing from low-resolution video," *Signal Process.*, vol. 91, pp. 1771–1780, 2011.

DOI:10.15774/PPKE.ITK.2011.005

[31] W. Carrington, K. Fogarty, and F. Fay, *3D fluorescence imaging of single cells using image restoration*. New York: Wiley-Liss, USA: K. Foskett and S. Grinstein, 1990.

[32] R. L. Lagendijk and J. Biemond, *Iterative identification and restoration of images*. Kluwer Academic Publishers, 1991.

[33] H. T. M. Vandervoort and K. C. Strasters, "Restoration of confocal images for quantitative image-analysis," *Journal of MicroscopyOxford*, vol. 178, pp. 165–181, 1995.

[34] W. Carrington and K. Fogarty, "3-d molecular distribution in living cells by deconvolution of optical sections using light microscopy," *Proc 13th Ann Northe Bioeng Conf*, pp. 108–111, 1987.

[35] P. J. Verveer and T. M. Jovin, "Acceleration of the ictm image restoration algorithm," *Journal of Microscopy*, vol. 188, no. 188, pp. 191–195, 1997.

[36] P. J. Verveer and T. M. Jovin, "Improved restoration from multiple images of a single object: application to fluorescence microscopy.," *Applied Optics*, vol. 37, no. 26, pp. 6240–6246, 1998.

[37] P. Sarder and A. Nehorai, "Deconvolution methods for 3-d fluorescence microscopy images," *IEEE In Signal Processing Magazine*, vol. 23, no. 3, pp. 32–45, 2006.

[38] T. J. Holmes, "Maximum-likelihood image restoration adapted for noncoherent optical imaging," *Journal of the Optical Society of America A*, vol. 5, no. 5, pp. 666–673, 1988.

[39] T. J. Holmes and Y. H. Liu, "Richardson-lucy/maximum likelihood image restoration algorithm for fluorescence microscopy: further testing.," *Applied Optics*, vol. 28, no. 22, pp. 4930–4938, 1989.

[40] T. J. Holmes and Y. H. Liu, "Acceleration of maximum-likelihood image-restoration for fluorescence microscopy and other noncoherent imagery," *Journal of the Optical Society of America AOptics Image Science and Vision*, vol. 8, no. 6, pp. 893–907, 1991.

DOI:10.15774/PPKE.ITK.2011.005

[41] T. J. Holmes, "Blind deconvolution of quantum-limited incoherent imagery: maximum-likelihood approach.," *Journal of the Optical Society of America A Optics and image science*, vol. 9, no. 7, pp. 1052–1061, 1992.

[42] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society Series B Methodological*, vol. 39, no. 1, pp. 1–38, 1977.

[43] G. M. P. Van Kempen, L. J. Van Vliet, P. J. Verveer, and H. T. M. Van Der Voort, "A quantitative comparison of image restoration methods for confocal microscopy," *Journal of Microscopy*, vol. 185, no. 3, pp. 354–365, 1997.

[44] D. L. Snyder, A. M. Hammoud, and R. L. White, "Image recovery from data acquired with a charge-coupled-device camera.," *Journal of the Optical Society of America A Optics and image science*, vol. 10, no. 5, pp. 1014–1023, 1993.

[45] S. Joshi and M. I. Miller, "Maximum Îą posteriori estimation with good's roughness for three-dimensional optical-sectioning microscopy," *J. Opt. Soc. Am.*, vol. 10, pp. 1078–1085, 1993.

[46] J. A. Conchello and J. McNally, "Fast regularization technique for expectation maximization algorithm for optical sectioning microscopy," *International Society for Optical*, 1996.

[47] J. Markham and J.-A. Conchello, "Parametric blind deconvolution: a robust method for the simultaneous estimation of image and blur," *Journal of the Optical Society of America A*, vol. 16, pp. 2377–2391, 1999.

[48] G. R. Ayers and J. C. Dainty, "Iterative blind deconvolution method and its applications," *Opt. Lett.*, vol. 13, no. 7, pp. 547–549, 1988.

[49] Y.-H. Liu, V. Krishnamurthi, S. Bhattacharyya, J. N. Turner, and T. J. Holmes, "Blind deconvolution of fluorescence micrographs by maximum-likelihood estimation," *Applied Opt.*, vol. 34, pp. 6633–6647, 1995.

[50] D. S. C. Biggs and M. Andrews, "Acceleration of iterative image restoration algorithms," *Appl. Opt*, vol. 36, no. 8, pp. 1766–1775, 1997.

DOI:10.15774/PPKE.ITK.2011.005

[51] R. W. R.J. Hanisch and R. Gilliland, "Deconvolutions of Hubble space telescope images and spectra," *in "Deconvolution of Images and Spectra", Ed. P.A. Jansson, 2nd ed., Academic Press, CA,*, 1997.

[52] J. Verbeeck and G. Bertoni, "Deconvolution of core electron energy loss spectra," *Ultramicroscopy*, vol. 109, no. 11, pp. 1343 – 1352, 2009.

[53] N. Dey, L. Blanc-Fraud, C. Zimmer, Z. Kam, P. Roux, J. Olivo-Marin, and J. Zerubia, "Richardson-lucy algorithm with total variation regularization for 3d confocal microscope deconvolution," *Microscopy Research Technique*, vol. 69, pp. 260–266, 2006.

[54] Y.-W. Wen and A. M. Yip, "Adaptive parameter selection for total variation image deconvolution," *Numer. Math. Theor. Meth. Appl.*, vol. 2, pp. 427–438, 2009.

[55] H. C. Andrews and B. R. Hunt, *Digital Image Restoration*. Prentice Hall Professional Technical Reference, 1977.

[56] G. M. P. V. Kempen, "Image restoration in fluorescence microscopy," *Microscopy*, p. 161, 1999.

[57] A. Papoulis, *Probability, Random Variables ad Stochastic Processes*. New York: McGraw-Hills, 1984.

[58] S. Osher, A. Sole, and L. Vese, "Image decomposition and restoration using total variation minimization and the h^{-1} norm," *Sci. Multiscale Model. Simul*, vol. 1, pp. 349–370, 2002.

[59] S. A. Shafer, "Color," ch. Using color to separate reflection components, pp. 43–51, USA: Jones and Bartlett Publishers, Inc., 1992.

[60] L. Havasi, Z. Szlavik, and T. Sziranyi, "The use of vanishing point for the classification of reflections from foreground mask in videos," *IEEE Transactions on Image Processing*, vol. 18, no. 6, pp. 1366 –1372, 2009.

[61] A. Kiss and T. Sziranyi, "Reconstructing static scene viewed through smoke using video," in *IEEE International Conference on Image Processing*, 2011.

DOI:10.15774/PPKE.ITK.2011.005

[62] R. Cucchiara, C. Grana, A. Prati, and R. Emilia, "Detecting moving objects and their shadows: An evaluation with the pets2002 dataset," in *PETS02*, pp. 18–25, 2002.

[63] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," vol. 25, no. 10, pp. 1337–1342, 2003.

[64] C. Benedek and T. Sziranyi, "Bayesian foreground and shadow detection in uncertain frame rate surveillance videos.," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 608–621, 2008.

[65] J.-L. Starck, M. Elad, and D. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1570 –1582, 2005.

[66] Y. Meyer, *Oscillating Patterns in Image Processing and Nonlinear Evolution Equations: The Fifteenth Dean Jacqueline B. Lewis Memorial Lectures*. Boston, MA, USA: American Mathematical Society, 2001.

[67] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577–685, 1989.

[68] F. Malgouyres, "Combining total variation and wavelet packet approaches for image deblurring," in *IEEE Workshop on Variational and Level Set Methods in Computer Vision*, 2001.

[69] F. Malgouyres, "Mathematical analysis of a model which combines total variation and wavelet for image restoration1," *Journal of Information Processes*, 2002.

[70] G. Hewer, W. Kuo, G. Hanson, and F. Sickman, "Double density complex wavelet based image cartoon-texture decomposition," in *Signals, Systems and Computers, 2006. ACSSC '06. Fortieth Asilomar Conference on*, pp. 861 – 868, 2006.

DOI:10.15774/PPKE.ITK.2011.005

[71] S. Roudenko, "Noise and texture detection in image processing," Tech. Rep. LANL report W-7405-ENG-36, Arizona State University, 2004.

[72] P. Comon, "Independent component analysis, a new concept?," *Signal Process.*, vol. 36, pp. 287–314, 1994.

[73] M. Zibulevsky and B. A. Pearlmutter, "Blind Source Separation by Sparse Decomposition in a Signal Dictionary," *Neural Comp.*, vol. 13, no. 4, pp. 863–882, 2001.

[74] J.-L. Starck, M. Elad, and D. Donoho, "Redundant multiscale transforms and their application for morphological component analysis," *Advances in Imaging and Electron Physics*, vol. 132, 2004.

[75] J. l. Starck, M. Elad, and D. Donoho, "Image decomposition via the combination of sparse representations and a variational approach," *IEEE Transactions on Image Processing*, vol. 14, pp. 1570–1582, 2005.

[76] M. Elad, J. Starck, P. Querre, and D. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA)," *Applied and Computational Harmonic Analysis*, vol. 19, no. 3, pp. 340–358, 2005.

[77] J. Bobin, Y. Moudden, J. l. Starck, and M. Elad, "Morphological diversity and source separation," *IEEE Signal Processing Letters*, vol. 13, pp. 409–412, 2006.

[78] J. Bobin, J. luc Starck, J. M. Fadili, Y. Moudden, and D. L. Donoho, "Morphological component analysis: An adaptive thresholding strategy," *IEEE Transactions on Image Processing*, vol. 16, pp. 2675–2681, 2007.

[79] J.-F. Aujol and G. Gilboa, "Constrained and snr-based solutions for tv-hilbert space image denoising," *J. Math. Imaging Vis.*, vol. 26, pp. 217–237, 2006.

[80] G. Sapiro and D. Ringach, "Anisotropic diffusion of multivalued images with applications to color filtering," *IEEE Transactions on Image Processing*, vol. 5, no. 11, pp. 1582 –1586, 1996.

DOI:10.15774/PPKE.ITK.2011.005

[81] J. Weickert, *Anisotropic Diffusion in Image Processing*. Stuttgart, Germany: Teubner-Verlag, 1998.

[82] D. Szolgay and T. Sziranyi, "Optimal stopping condition for iterative image deconvolution by new orthogonality criterion," *Electronics Letters*, vol. 47, no. 7, pp. 442–444, 2011.

[83] T. Lindeberg, "Edge detection and ridge detection with automatic scale selection," *Int. Journal of Computer Vision*, vol. 30, pp. 117–154, 1998.

[84] D. Gabor, "Information theory in electron microscopy," *Laboratory Investigation*, vol. 14/6, pp. 801–807, 1965.

[85] L. Florack, *Image Structure*. Kluwer Academic Publishers, 1997.

[86] L. Alvarez, P.-L. Lions, and J.-M. Morel, "Image selective smoothing and edge detection by nonlinear diffusion. ii," *SIAM J. Numer. Anal.*, vol. 29, pp. 845–866, 1992.

[87] I. Kopilovic and T. Sziranyi, "Artifact reduction with diffusion preprocessing for image compression," *Optical Engineering*, vol. 44, no. 2, 2005.

[88] S. Mann, "Wearable computing: a first step toward personal imaging," *Computer*, vol. 30, no. 2, pp. 25 –32, 1997.

[89] "Memory and sharing of experiences," *Personal Ubiquitous Comput.*, vol. 11, no. 4, 2007.

[90] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," vol. 2, pp. 28 – 31 Vol.2, 2004.

[91] Z. Zivkovic and F. van der Heijden, "Recursive unsupervised learning of finite mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 651 –656, 2004.

[92] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recogn. Lett.*, vol. 27, pp. 773–780, 2006.

DOI:10.15774/PPKE.ITK.2011.005

[93] L. Carminati and J. Benois-Pineau, "Gaussian mixture classification for moving object detection in video surveillance environment," vol. 3, pp. III – 113–16, 2005.

[94] M. Balcells, D. DeMenthon, and D. Doermann, "An appearance-based approach for consistent labeling of humans and objects in video," *Pattern Anal. Appl.*, vol. 7, pp. 373–385, 2004.

[95] K. Kim, T. Chalidabhongse, D. Harwood, and L. Davis, "Background modeling and subtraction by codebook construction," vol. 5, pp. 3061 – 3064 Vol. 5, 2004.

[96] T. Tian, C. Tomasi, and D. Heeger, "Comparison of approaches to egomotion computation," pp. 315 –320, 1996.

[97] G. L. Foresti and C. Micheloni, "A robust feature tracker for active surveillance of outdoor scenes," *Electronic Letters on Computer Vision and Image Analysis*, pp. 21–34, 2003.

[98] B. Jung and G. S. Sukhatme, "Detecting moving objects using a single camera on a mobile robot in an outdoor environment," pp. 980–987, 2004.

[99] J.-H. Ahn, C. Choi, S. Kwak, K. Kim, and H. Byun, "Human tracking and silhouette extraction for human-robot interaction systems," *Pattern Anal. Appl.*, vol. 12, pp. 167–177, 2009.

[100] T. Veit, F. Cao, and P. Bouthemy, "An a contrario decision framework for region-based motion detection," *Int. J. Comput. Vision*, vol. 68, pp. 163–178, 2006.

[101] C. Yuan, G. Medioni, J. Kang, and I. Cohen, "Detecting motion regions in the presence of a strong parallax from a moving camera by multiview geometric constraints," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, pp. 1627 –1641, 2007.

[102] Y. Sheikh and M. Shah, "Bayesian modeling of dynamic scenes for object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1778 –1792, 2005.

DOI:10.15774/PPKE.ITK.2011.005

[103] I. Kartika and S. Mohamed, "Frame differencing with post-processing techniques for moving object detection in outdoor environment," in *International Colloquium on Signal Processing and its Applications (CSPA)*, pp. 172 –176, 2011.

[104] B. Lo and S. Velastin, "Automatic congestion detection system for underground platforms," in *International Symposium on Intelligent Multimedia, Video and Speech Processing,*, pp. 158 –161, 2001.

[105] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780 –785, 1997.

[106] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," vol. 2, pp. 246–252, IEEE Computer Society, 1999.

[107] J. Zhang and C. H. Chen, "Moving objects detection and segmentation in dynamic video backgrounds," in *IEEE Conference on Technologies for Homeland Security,*, pp. 64 –69, 2007.

[108] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric Model for Background Subtraction," vol. 1843 of *Lecture Notes in Computer Science*, pp. 751–767, Springer Berlin Heidelberg, 2000.

[109] A. Elgammal, R. Duraiswami, D. Harwood, and L. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," *Proceedings of the IEEE*, vol. 90, no. 7, pp. 1151 – 1163, 2002.

[110] B. Han, D. Comaniciu, and L. Davis, "Sequential kernel density approximation through mode propagation: Applications to background modeling," in *Asian Conference on Computer Vision*, 2004.

[111] S. Berrabah, G. De Cubber, V. Enescu, and H. Sahli, "Mrf-based foreground detection in image sequences from a moving camera," in *IEEE International Conference on Image Processing,*, pp. 1125 –1128, 2006.

DOI:10.15774/PPKE.ITK.2011.005

[112] K. Peres, C. Helmer, H. Amieva, J.-M. Orgogozo, I. Rouch, J.-F. Dartigues, and P. Barberger-Gateau, "Natural history of decline in instrumental activities of daily living performance over the 10 years preceding the clinical diagnosis of dementia: A prospective population-based study," *Journal of the American Geriatrics Society*, vol. 56, no. 1, pp. 37–44, 2008.

[113] R. Megret, D. Szolgay, J. Benois-Pineau, P. Joly, J. Pinquier, J.-F. Dartigues, and C. Helmer, "Wearable video monitoring of people with age dementia: Video indexing at the service of healthcare," in *International Workshop on Content-Based Multimedia Indexing, 2008*, pp. 101 – 108, 2008.

[114] B. W. Silverman, *Density estimation: for statistics and data analysis*. Chapman and Hall, London ; New York, 1986.

[115] E. Fix and J. L. Hodges, "Discriminatory Analysis: Nonparametric Discrimination: Consistency Properties," Tech. Rep. Project 21-49-004, Report Number 4, USAF School of Aviation Medicine, Randolph Field, TX, 1951.

[116] M. Rosenblatt, "Remarks on Some Nonparametric Estimates of a Density Function," *The Annals of Mathematical Statistics*, vol. 27, no. 3, pp. 832–837, 1956.

[117] D. W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization (Wiley Series in Probability and Statistics)*. Wiley, 1992.

[118] G. R. Terrell and D. W. Scott, "Variable kernel density estimation," *Annals of Statistics*, vol. 20, pp. 1236–1265, 1992.

[119] M. C. Minnotte, "A test of mode existence with applications to multimodality," 1992. Unpublished Ph.D. Dissertation, Dept. Statistics, Rice Univ.

[120] D. O. Loftsgaarden and C. P. Quesenberry, "A nonparametric estimate of a multivariate density function," *Ann. Math. Statist.*, vol. 36, no. 3, pp. 1049–1051, 1965.

[121] Y. P. Mack and M. Rosenblatt, "Multivariate k-nearest neighbor density estimates," *Journal of Multivariate Analysis*, vol. 9, no. 1, pp. 1–15, 1979.

DOI:10.15774/PPKE.ITK.2011.005

[122] "On near neighbour estimates of a multivariate density," *Journal of Multivariate Analysis*, vol. 13, no. 1, pp. 24 – 39, 1983.

[123] L. Breiman, W. Meisel, and E. Purcell, "Variable kernel estimates of multivariate densities," *Technometrics*, vol. 19, no. 2, pp. 135–144, 1977.

[124] I. S. Abramson, "Arbitrariness of the pilot estimator in adaptive kernel methods," *Journal of Multivariate Analysis*, vol. 12, no. 4, pp. 562 – 567, 1982.

[125] S. Sain, "Adaptive Kernel Density Estimation," 1994. Unpublished Ph.D. Dissertation, Department of Statistics, Rice University.

[126] V. A. Epanechnikov, "Non parametric estimation of a multivariate probability density.," *Theory of Probability and Its Applications*, no. 14, pp. 153–158, 1969.

[127] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, 1999.

[128] P. H. A. Sneath and R. R. Sokal, "Numerical taxonomy: the principles and practice of numerical classification," in *Computer Animation Conference*, 1973.

[129] B. King, "Step-Wise Clustering Procedures," *Journal of the American Statistical Association*, vol. 62, no. 317, pp. 86–101, 1967.

[130] J. H. Ward, "Hierarchical Grouping to Optimize an Objective Function," *Journal of the American Statistical Association*, vol. 58, no. 301, pp. 236–244, 1963.

[131] F. Murtagh, "A survey of recent advances in hierarchical clustering algorithms which use cluster centers," *Comput. J.*, vol. 26, pp. 354–359, 1984.

[132] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297, University of California Press, 1967.

DOI:10.15774/PPKE.ITK.2011.005

[133] A. K. Jain and R. C. Dubes, *Algorithms for clustering data*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1988.

[134] T. Mitchell, *Machine Learning (Mcgraw-Hill International Edit)*. McGraw-Hill Education (ISE Editions), 1st ed., 1997.

[135] S.-Y. Lu and K. S. Fu, "A sentence-to-sentence clustering procedure for pattern analysis," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 8, no. 5, pp. 381 –389, 1978.

[136] L. Zadeh, "Fuzzy sets," *Information Control*, vol. 8, pp. 338–353, 1965.

[137] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1027–1035, Society for Industrial and Applied Mathematics, 2007.

[138] J. Bergen, P. Burt, R. Hingorani, and S. Peleg, "A three-frame algorithm for estimating two-component image motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence,*, vol. 14, no. 9, pp. 886 –896, 1992.

[139] S. Ayer and H. Sawhney, "Layered representation of motion video using robust maximum-likelihood estimation of mixture models and mdl encoding," in *In Proceedings of the Fifth International Conference on Computer Vision,*, pp. 777 –784, 1995.

[140] Y. Aloimonos, ed., *Active Perception*. Hillsdale, NJ, USA: L. Erlbaum Associates Inc., 1993.

[141] J. Costeira and T. Kanade, "A multi-body factorization method for motion analysis," in *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pp. 1071 –1076, 1995.

[142] J. Lawn and R. Cipolla, "Robust egomotion estimation from affine motion parallax," in *European Conference on Computer Vision,*, vol. 800 of *Lecture Notes in Computer Science*, pp. 205–210, Springer Berlin / Heidelberg, 1994.

DOI:10.15774/PPKE.ITK.2011.005

[143] M. Irani and P. Anandan, "A unified approach to moving object detection in 2d and 3d scenes," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 6, pp. 577 –589, 1998.

[144] M. Accame, F. G. B. D. Natale, and D. D. Giusto, "High performance hierarchical block-based motion estimation for real-time video coding," *Real-Time Imaging*, vol. 4, no. 1, pp. 67 – 79, 1998.

[145] C. Archambeau, M. Valle, A. Assenza, and M. Verleysen, "Assessment of probability density estimation methods: Parzen window and finite gaussian mixtures," p. 4 pp., 2006.

[146] I. S. Abramson, "On bandwidth variation in kernel estimates - a square root law," *Ann. Statist.*, vol. 10, no. 10, pp. 1217–1223, 1982.

[147] B. J. Worton, "Optimal smoothing parameters for multivariate fixed and adaptive kernel methods," *Journal of Statistical Computation and Simulation*, vol. 32, pp. 45 – 57, 1989.

[148] A. Bugeau, *Détection et suivi d'objets en mouvement dans des scénes complexes, application á la surveillance des conducteurs*. Thése de l'université de Rennes 1, Mention Traitement du Signal et des Télécommunications, 2007.

[149] J. Sander, M. Ester, H.-P. Kriegel, and X. Xu, "Density-based clustering in spatial databases: The algorithm gdbscan and its applications," *Data Min. Knowl. Discov.*, vol. 2, pp. 169–194, 1998.

[150] C. J. van Rijsbergen, *Information Retrieval*. Butterworth-Heinemann, London, 2nd edition, 1979.

[151] Z. Zivkovic, "Zoran Zivkovic's home page." `http://staff.science.uva.nl/~zivkovic/DOWNLOAD.html`.

[152] G. Gupta and C. Chakrabarti, "Architectures for hierarchical and other block matching algorithms," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, no. 6, pp. 477 –489, 1995.

DOI:10.15774/PPKE.ITK.2011.005

[153] S. Mazare, R. Pacalet, and J.-L. Dugelay, "Using gpu for fast block-matching," in *In European Signal Processing Conference, Florence, Italy*, 2006.

DOI:10.15774/PPKE.ITK.2011.005

Publications of the Author

D. Szolgay, J. Benois-Pineau, R. Megret, Y. Gaestel, and J.-F. Dartigues, "Detection of moving foreground objects in videos with strong camera motion," *Pattern Analysis and Applications*. accepted in 04.04.2011.

D. Szolgay and T. Sziranyi, "Optimal stopping condition for iterative image deconvolution by new orthogonality criterion," *Electronics Letters*, vol. 47, no. 7, pp. 442–444, 2011.

D. Szolgay and T. Sziranyi, "Adaptive image decomposition into cartoon and texture parts optimized by the orthogonality criterion," *IEEE Transactions on Image Processing*. Submitted in May 2011.

D. Szolgay, C. Benedek, and T. Sziranyi, "Fast template matching for measuring visit frequencies of dynamic web advertisements," *Proceedings of VISAPP 2008, Third International Conference on Computer Vision Theory and Applications.*, pp. 228–233, 2008.

R. Megret, D. Szolgay, J. Benois-Pineau, P. Joly, J. Pinquier, J.-F. Dartigues, and C. Helmer, "Wearable video monitoring of people with age dementia : Video indexing at the service of health care," in *International Workshop on Content-Based Multimedia Indexing, 2008.*, pp. 101 – 108, 2008.

D. Szolgay and T. Szirányi, "Orthogonality based stopping condition for iterative image deconvolution methods," in *Computer Vision ACCV 2010*, vol. 6495 of *Lecture Notes in Computer Science*, pp. 321–332, Springer Berlin / Heidelberg, 2011.