Incorporating Near-Infrared into Scene Understanding

THÈSE N^{O} 5806 (2013)

PRÉSENTÉE LE 28 JUIN 2013

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS

GROUPE IMAGES ET REPRÉSENTATION VISUELLE

PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Neda SALAMATI

acceptée sur proposition du jury:

Prof. P. Fua, président du jury
Prof. S. Süsstrunk, directrice de thèse
Dr G. Csurka, rapporteur
Prof. R. Hersch, rapporteur
Prof. S. Marchand-Maillet, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2013

To my parents...

Acknowledgements

Over the last few years, I have been lucky to work at EPFL with a fantastic supervisor and colleagues all the while living in beautiful Lausanne. I can hardly think of a better place to do research.

Foremost, I would like to express my sincere gratitude to my advisor Prof. Sabine Süsstrunk for the continuous support of my Ph.D study and research, for her patience, motivation, and immense knowledge. Her guidance was invaluable, but more importantly, her encouragement, support and invaluable advice boosted my confidence and helped me succeed throughout one of the most difficult years in my personal life.

I would like to express my deepest gratitude to Dr. Michael Brill for believing in me, inspiring me, and giving me the confidence to pursue my goals with hard work and dedication. I believe my success is at least in part due to his sincere support and guidance.

Further, I would like to thank Gabriela Csurka and Diane Larlus for their hospitality and the time we worked together. It was a great experience to be able to visit XRCE in Grenoble, France. Also thanks to all the great people I met there. Gabriela and Diane have always given me a lot of encouragement, which has been very valuable for me. They have always given me detailed comments on ideas and results. Also thanks to Nelly Afonso for her contribution in acquiring the RGB+NIR database.

It was an honor to have a jury of distinguished researchers on my doctoral committee: Prof. Stéphane Marchand-Maillet, Dr. Gabriela Csurka, Prof. Roger Hersch, and Prof. Pascal Fua. I thank them for their time reading my thesis, the interesting discussion and all the valuable feedback.

Besides my supervisors, I am also grateful to many fellow students and colleagues who have come and gone during the last years at EPFL. Special thanks to Appu who has been there for me with a motivational speech and a cup of coffee, or two, every time I felt overwhelmed. I would like to heartily thank Cheryl who cheered me up every time I was feeling down, and Zahra for always being there for me when I needed a friend to talk. I would also like to thank Jan who read parts of my thesis, and helped in ordering my chaotic thoughts into clear writing. Also a big thank to Richa and Anshuman for being there for me when I needed their help the

most and helping me get back on my feet - thanks!
Finally, I would like to thank my friends and family from Iran whose tireless and unconditional support has been invaluable over the last few years. Thanks to all!

Lausanne, 2013

Abstract

Recent progress in computational photography has shown that we can acquire near-infrared (NIR) information in addition to the normal visible (RGB) information with only slight modification to the standard digital camera. In this thesis, we study if this extra channel can improve one of the more difficult computer vision tasks: Scene understanding. Due to the proximity of NIR to visible radiation, NIR images share many properties with visible images. However, as a result of the material dependency of reflection in the NIR part of the spectrum, such images reveal different characteristics of the scene. In this work we study how to effectively exploit these differences to improve scene recognition and semantic segmentation performance.

An initial psycho-physical test that we carried out gave promising evidence that humans understand the content of a scene more effectively when presented with the NIR image as opposed to the visible image. Motivated by this, we first formulate a novel framework that incorporates NIR information into a low-level segmentation algorithm to better detect the material boundaries of an object. This goal is achieved by first forming a illumination-invariant representation, i.e., the intrinsic image, and then by employing the material dependent properties of NIR images. Secondly, by leveraging on state-of-the-art segmentation frameworks and a novel manually segmented image database, we study how to best incorporate the specific characteristics of the NIR response into high-level semantic segmentation tasks.

We show through extensive experimentation that introducing NIR information significantly improves the performance of automatic labeling for certain object classes, like fabrics and water, whose response in the NIR domain is particularly discriminant. We then thoroughly discuss the results with respect to both physical properties of the NIR response and the characteristics of the segmentation framework.

Keywords: Scene understanding, Near-infrared imaging, Semantic segmentation, Image classification, Boundary detection, Material-based segmentation, CRF model, Graph-cut segmentation.

Zusammenfassung

Neuerungen auf dem Gebiet der computergestützten Bildgebung erlauben es uns, mit einer nur wenig modfizierten herkömmlichen Digitalkamera Aufnahmen nicht nur im üblichen RGB-Bereich, sondern auch im nahen Infrarot (NIR) zu machen. Diese Doktorarbeit untersucht, inwiefern wir diese zusätzliche Informationsquelle für eins der schwierigeren Probleme des Maschinellen Sehens nutzen können: automatisches Szeneverständnis.

NIR-Aufnahmen haben aufgrund der Nähe zum sichtbaren Bereich des Lichts viele Gemeinsamkeiten mit konventionellen RGB-Bildern, jedoch enthüllen sie aufgrund der stärker materialabhängigen Reflexion zusätzliche Szenecharakteristika. Wir untersuchen, wie wir diese am besten zum automatischem Szeneverständnis und zur semantischen Segmentierung verwerten können.

Durch eine einführende Studie können wir belegen, daß Menschen eine Szene leichter in der NIR-Repräsentation als im sichtbaren Spektralbereich erfassen. Dies motiviert uns erstens, ein Framework zu entwerfen, das NIR Informationen in einen low-level Segmentierungsalgorithmus einbindet, um zuerst ein beleuchtungsunabhängiges Bild der Szene zu erstellen und dann die Materialabhängigkeit der NIR-Aufnahme ausnutzen, und so Materialgrenzen in und zwischen Objekten besser zu finden. Zweitens untersuchen wir mithilfe eines Segmentierungsframeworks vom aktuellen dem Stand der Technik und einer neu erstellten Datenbank handsegmentierter Bilder Möglichkeiten zur Nutzung der NIR-spezifischen Bildcharakteristika zur high-level-Segmentierung.

Wir zeigen mit Hilfe ausgedehnter Experimentreihen, daß NIR-Information die Genauigkeit automatischer Etikettierung für manche Objektklassen, beispielsweise Textilien und Wasser, deren NIR-Reflexionsverhalten besonders gut unterscheidbar ist, deutlich erhöht. Schließlich diskutieren und deuten wir unsere Ergebnisse ausgiebig, mit besonderem Augenmerk sowohl auf die physikalischen Eigenschaften der NIR-Bildgebung als auch auf die Charakteristika des verwendeten Segmentierungsframeworks.

Schlagworte: Szeneverständnis, Nahes Infrarot, semantische Segmentierung, Bildkassifikation, Randerkennung, materialbasierte Segmentierung, CRF Modell, Segmentierung mittels Graphenschnitt.

Résumé

Les récents progrès en photographie computationelle ont démontré que nous pouvons acquérir l'information de l'infrarouge proche (PIR) en plus de l'information visible normal (RVB) en apportant juste une légère modification à un appareil photographique numérique standard. En raison de la proximité de PIR avec le spectre visible, les images PIR partagent de nombreuses propriétés avec les images classiques. Cependant, du fait de l'impact du matériau sur les réflections dans le proche infra rouge, ces images révèlent différentes caractéristiques de la scène. Dans cette thèse, nous étudions comment exploiter ces propriétés pour améliorer la performance d'algorithmes de segmentation sémantique et de reconnaissance de scène.

Une expérience psycho-physique initialement effectuée a démontré que les êtres humains comprennent mieux le contenu d'une scène avec une image infrarouge qu'avec une image normale. Cela nous a motivé à formuler une nouvelle approche qui incorpore l'information PIR dans un algorithme de segmentation bas niveau pour mieux détecter les limites des différent matériaux composant un objet. Ce but est atteint tout d'abord par la formation de l'image intrinsèque puis par l'exploitation des propriétés liées aux matériaux dans les images PIR. Nous étudions ensuite la meilleure maniére pour incorporer les caractéristiques spécifiques de la réponse PIR dans la segmentation sémantique haut-niveau en prenant avantage des méthodes de segmentation modernes et d'une nouvelle base de données d'images segmentées manuellement.

Nous montrons à travers de nombreuses expériences que l'introduction du PIR améliore de manière significative la performance des algorithmes de classification pour certaines catégories comme les tissus ou l'eau qui ont des réponses en PIR très différentes. Finalement, nous discutons en détails des résultats liés aux propriétés physiques du PIR ainsi que des caractéristiques de la méthode de segmentation.

Mots-clés : Reconnaissance de scène, Infrarouge proche imagerie, Segmentation sémantique, Classification des images, Segmentation dépendante des matériaux, Detection des bords, Model CRF, Segmentation avec Graph-cut.

Contents

List of Figures

List of Tables

1 Introduction

Figure 1.1: Examples from RGB-NIR images. Notice that the NIR band exhibits noticeable differences at the scene level.

Have you ever wondered what it would be like to be able to see near-infrared light, like an African fish or boa constrictor can? It is a whole new way of looking at the world. This thesis shows how near-infrared can help us to extract more accurate information about a scene.

One of the long-term goals in machine vision is to develop autonomous systems that can reason out the visual environment from visual inputs. Humans with proper prior knowledge "understand" a scene they see and could answer questions about the scene regarding the presence and locations of different objects. This is an easy task for a human being, as we are generally able to discard the influence of lighting conditions and hence to accurately identify the class (or object) that an image region belongs to [Finlayson et al., 1994]. Scene understanding is useful in many applications, including robot perception for navigation and surveillance for physical security and environmental monitoring. In recent years, we have seen much progress using sophisticated (or rather high-level) image descriptors [Sivic and Zisserman, 2003, Csurka et al., 2004, Perronnin and Dance, 2007] and better machine learning techniques [Boykov and Kolmogorov, 2004], although scene understanding [Li-Jia et al., 2009], including semantic segmentation [Shotton et al., 2006, Verbeek and Triggs, 2007, Ladicky et al., 2010, Csurka and Perronnin, 2011], still remains as a challenging task.

One reason that machine vision systems still struggle to understand scenes is mainly due to the ambiguity of the influence of light and surface reflectance on a given pixel value. For example, a dark pixel can either result from a dark surface reflectance under normal lighting conditions or a light surface reflectance under a shadow. More specifically, such shortcomings are mostly due to four elements - appearance variation, cluttered backgrounds, illumination differences, and shadow effects. Decoding the contributions of such effects from a scene is a hard problem [Finlayson et al., 1994]. To solve it, we either need to make assumptions about the world or to capture more information.

In this thesis, we study scene understanding using the second approach. Specifically, we propose to use near-infrared (NIR) images, in addition to visible (RGB) images, as input to the scene understanding task. NIR is electromagnetic radiation at wavelengths that range from 700 to 1100 nanometers, just beyond the red part of the visible spectrum. Due to the proximity of NIR to visible radiation, NIR images share many properties with visible images. However, as a result of the material dependency of reflection in the NIR part of the spectrum, such images reveal different characteristics of the scene [Salamati et al., 2009, Salamati and Süsstrunk, 2010] (see Figure 1.2 for illustration).

NIR information can potentially be captured by any digital camera. Silicon sensors of standard digital cameras are naturally sensitive in the NIR (700-1100nm) wavelength range. By removing the hot mirror (a NIR blocking filter) affixed to the sensor, digital cameras can be enabled to capture both visible and NIR images [Fredembach and Süsstrunk, 2008]. Combining both scene representations has recently been successfully used in a number of computer vision and

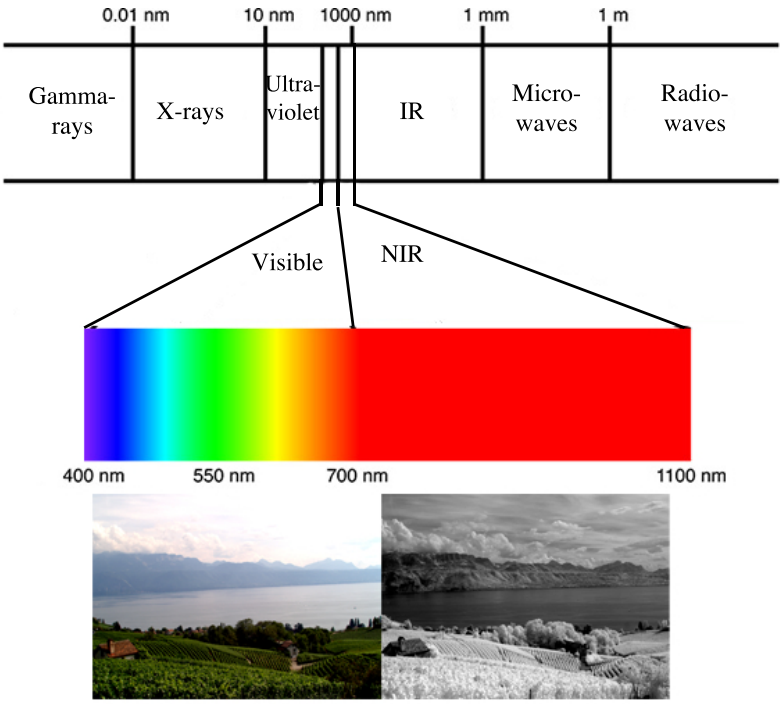


Figure 1.2: Electromagnetic spectrum, and a scene in both the visible and NIR part of the spectrum.

computational photography applications, such as dehazing [Schaul et al., 2009], dark flash photography [Krishnan and Fergus, 2009], skin smoothing [Süsstrunk et al., 2010], and scene categorization [Brown and Süsstrunk, 2011, Salamati et al., 2011b].

The properties of NIR images have been used by the remote sensing [Zhou et al., 2009, Walter, 2004] and military [Kong et al., 2005] communities for many years to detect and classify natural and/or man-made objects, with a focus on aerial photography and human detection. These applications typically use true hyper-spectral capture with several bands in the NIR and also the IR part of the spectrum. However, in this thesis we present a framework that uses only a single channel for integrating all NIR radiation. The single channel NIR can be easily captured by a standard sensor found in any digital camera. This allows us to tackle scene understanding challenges in everyday photography.

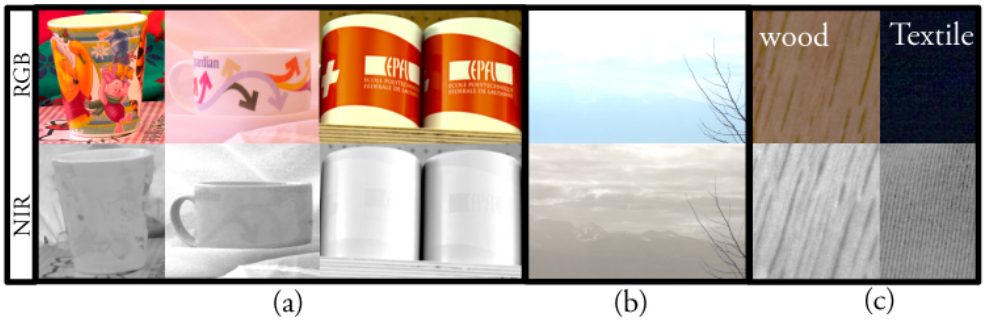


Figure 1.3: Challenging cases for RGB-only semantic segmentation. In NIR images (a) cups in different colors share the same brightness, (b) haze is transparent, and (c) texture is more intrinsic to the material.

1.1 NIR Image Characteristics

Intrinsic properties of the NIR wavelength band guarantee that images can be sharper, less affected by man-made colorants, and more resilient to changing light conditions. Exploiting the characteristics of the different waveband images can lead to improved scene understanding and object recognition. In this section, we review some physical phenomena that result in differences between RGB and NIR images.

1. **Rayleigh scattering** When light is scattered on very small particles ($s < \lambda/10$), it behaves according to Rayleigh scattering, $E_s \propto E_0/\lambda^4$, i.e., the intensity of the scattered light E_s is proportional to that of the incident light E_0 by the inverse of the fourth power of the wavelength λ. As a result,

    - Sky appears blue in an RGB image because it is the most scattered color due to its relatively short wavelength. In comparison, NIR at 1000 nm is 40 times less scattered than blue at 400nm. We thus expect NIR intensity to be significantly lower than its visible counterpart in sky and its reflected components, such as water.
    - Haze is transparent in NIR images. Haze is caused by scattering on particles ($s < \lambda/10$) in the air. Hence, scattering follows Rayleigh law, and, therefore, haze is almost not present in the NIR.

    Using RGB-only images often yields mis-recognition of mountains, sky, and clouds in hazy atmosphere (see Figure 1.3-b for illustration). The “haze transparency” characteristic of NIR results in sharper images for distant objects. In particular, vegetation,

Figure 1.4: Automatic semantic segmentation results. (a) RGB images, (b) NIR images, (c) the results obtained by taking into account only RGB information, and (d) the results of incorporating NIR information as well as RGB. In the top image, you can notice that incorporating material-dependent NIR information results in a more accurate recognition. The high contrast between the sky and clouds in the NIR image of the scene in the bottom results in a more precise clouds boundary.

mountains, sky, and clouds at a distance in the visible image are smoothed and bluish, which may affect the performance of texture and color features in the classification task. The sharper and haze-free appearance of these classes in NIR helps classification and leads to better segmentation.

2. **A very large number of pigments are quasi-transparent to NIR radiation** [Burns and Ciurczak, 2001, Salamati et al., 2009, Fredembach and Süsstrunk, 2008] show that most of dyes and pigments have little or no absorption in the NIR. As a result,

   - NIR images reveal more information about the material. Due to the transparency of colors, the intensity values in NIR images are more consistent across a single material and, consequently, across a given class region (see Figure 1.3-a). NIR images provide information that can be used to automatically identify material classes.
   - Texture is more intrinsic to the material in NIR images. Due to the colorants' transparency, material intrinsic texture properties are easier to capture in the NIR part of the spectrum.

   Considering color in RGB images as a discriminative feature causes difficulties in situations where there is a cluttered background. Moreover, man-made classes, such as Screen, Cup, and Clothing, are made of a variety of colors and patterns (see Figure 1.3-a). Thus, learning an appearance model of such classes based on RGB values only is very challenging. Even for classes with a consistent color, such as Vegetation, relying on

solely RGB-based features can be insufficient as the color information perceived by a camera drastically varies in different lighting conditions.

Texture has also shown to be a powerful cue for semantic segmentation [Csurka and Perronnin, 2011, Plath et al., 2009]. This local feature, however, can be confused with color patterns and other immediate surface impurities, as surface reflectance in the visible part of the spectrum is not intrinsic to the corresponding object material (see Figure 1.3-c for illustration).

The ultimate goal of automatic scene recognition and object labeling is to design a model that produces the same segments and labels as humans do. Therefore, the results of computer vision algorithms are often evaluated against a ground truth annotated by a human. Thus, in order to design a more accurate model, we need to understand how humans assign semantic meanings to different regions of an image. There are several attributes that play a role in this task: color, texture, material, and context. Taking into account the characteristics of NIR images (material dependency and the fact that perceived texture is more intrinsic to the class of material), it is expected that NIR data improves the accuracy of automatic labeling for the cases in which the semantic class corresponds to a specific material, texture or combination of them. For instance, the class Cup is often made of very specific classes of material (porcelain, ceramic, or plastic) and, regardless of the pattern and color of the object, NIR has a unique reflection response for each of these materials. Consequently, we expect to increase the accuracy of recognition for such classes by incorporating NIR into the segmentation framework. The classes Building or Car, however, usually consist of a variety of man-made material classes. Thus, introducing NIR is not expected to significantly improve the performance of automatic labeling for them. In summary, incorporating NIR information is expected to outperform the visible-only strategies for the cases when the key attribute to assign a certain class to a region is

- texture (Fabric and Wood),
- material (Water, Cloud, and Vegetation), or
- a combination of texture and material.

For the cases where color is the key to recognizing a class, NIR is unlikely to significantly improve the accuracy.

In this thesis, we propose to incorporate NIR information, in addition to RGB images, into a state-of-the-art image classification, low-level and high-level segmentation framework to overcome the shortcomings of RGB-based approaches mentioned above. It has been shown that each class of material has an intrinsic behavior in the NIR part of the spectrum [Salamati et al., 2009]. See Figure 1.4 and notice that many sources of confusion and mis-segmentation can be avoided by incorporating NIR images. We specifically investigate if NIR information can increase the accuracy of human cognition, automatic image classification, and semantic segmentation. The specific characteristics of NIR are proposed to be helpful in all those tasks.

Human Cognition:
Visual cognition occurs mostly based on the shape properties of the objects rather than the color or patterns [Ullman and Power, 1997]. Since objects made of a specific material usually have the same response in NIR images, it is more probable that edges in NIR images represent the physical shape of the object rather than changes in color within the object. We therefore expect that NIR images are useful input for human scene cognition tasks and competetive with images in the visible spectrum. Verifying this allows us to easily demonstrate the general appropriateness of using NIR in scene understanding.

Automatic Scene Classification:
Automatic scene recognition is a long-standing problem in computer vision. It is an important element in contextual vision [Krishnan and Fergus, 2009, Fei-Fei et al., 2005]. In computer vision applications, the effective use of color for image classification mainly requires illumination estimation [Finlayson et al., 1994] or computing invariants [Geusebroek et al., 2001] under different illumination conditions. One attraction of incorporating NIR information is that it is less correlated with R, G and B than they are with each other, which should increase any gains from effective multispectral techniques.

Semantic Image Segmentation:
Semantic image segmentation is the process of partitioning an image into regions, where each region corresponds to a semantic class within a predefined list. We believe that the intrinsic properties of NIR images make them as relevant for the semantic segmentation task as conventional RGB images. First, NIR images share many characteristics with visible images, due to the NIR radiation being adjacent to the visible spectrum. In particular, the shapes of objects in the scene are preserved, i.e., borders of physical objects in the visible images match the borders in the NIR image, which is necessary for segmentation. Second, the intensity values in the NIR images are more consistent across a single material, and consequently across a given class region, due to the unique reflectances of certain natural and

man-made composites to NIR radiation [Salamati et al., 2009] Third, texture in NIR images is more intrinsic to the material. This is partly due to the transparency of most colorants and dyes in NIR; texture introduced by (color) patterns on the surface is less dominant in NIR. Additionally, there is generally less haze present in NIR images [Schaul et al., 2009].

1.2 Contributions

The results of this dissertation are structured into five main chapters. Here, we briefly describe the main contributions of each chapter.

Chapter 2: As a preliminary study, in this chapter, we investigate the potential offered by NIR images to more accurately classify different types of material classes:

- We acquire a dataset of visible and NIR images under controlled viewpoint and illumination conditions. The dataset consists of 51 samples wood, tile, textile, and linoleum samples.
- Image features are proposed according to the characteristics of NIR images and the relation with the visible images.
- We show that using the proposed features leads to a better classification of the material classes when NIR information is present.

Chapter 4: In this chapter we validate the hypothesis that humans can understand the content of a scene more effectively when presented with the NIR image as opposed to the visible image. To this end:

- We execute a psychophysical experiment to measure the human cognition threshold for both visible and NIR representations. The promising evidence from this experiment gives us enough belief that an automatic computer system can leverage NIR information for improved recognition accuracy as well.

Chapter 5: We consider the task of automatic image classification in the context of images for which both standard visible RGB channels and NIR information are available. We use an efficient local patch based image representation:

- Consistent with previous work, we confirm the observation that the combination of both color and NIR cues can be useful for the image categorisation task, in a state-of-the art framework.
- We conduct a thorough study on how to compute and best use texture and color descriptors when NIR information is available.
- We investigate the complementarity between the different descriptors considered, and propose efficient ways to combine them.

Chapter 6: We present a framework to incorporate NIR information in order to better segment an image into objects by separating material boundaries from color and shadow edges:

- We form an intrinsic image by extending the 4-sensor camera calibration model by [Finlayson and Drew, 2001] into incorporating the NIR channel along with RGB channels.
- We propose a low-level segmentation framework based on the idea that the union of both segmentations obtained from the intrinsic and NIR images results in image partitions that are only based on material changes and not on color or shadows.
- We present results, showing that the proposed method provides good object-based segmentation results on diverse images.

Chapter 7: We propose an approach to incorporate NIR information into semantic image segmentation:

- We contribute with a pixel-level annotation of a dataset of 770 registered RGB and NIR image pairs.
- Our second contribution is the extension of a state-of-the-art segmentation framework with different strategies for incorporating the NIR channel. Our proposed system is based on a "conditional random field" (CRF), where we exploit different possibilities for combining the visible and NIR information in the recognition part and in the regularization part of the model.
- In addition to the evaluation, we fully discuss the accuracy for each class of material, linked to the material characteristics of NIR radiation.

Chapter 8: In this chapter we present a summary of the thesis and discuss its contributions. We point towards the possibilities for improvement and the directions for future work.

2 Related Work in NIR Imaging

NIR spectra are influenced by the chemistry and physical structure of different material classes, which makes them suitable for material classification [Burns and Ciurczak, 2001]. Comparing NIR images to visible spectrum images, we observe several differences (see Figure 2.1 for illustration). The following are some interesting features of NIR images:

1. The sky is dark, while clouds are bright
2. Atmospheric haze disappears
3. Contrast is high
4. Vegetation is very bright
5. Patterns on some materials are transparent to NIR radiation

Properties (2) and (3) are closely related, because the haze transparency allows us to see the details behind it. Atmospheric haze transparency in the NIR part of the spectrum allows us to take haze-free landscape pictures even in difficult meteorologic conditions, or when the smog level is high enough to impair vision in the visible part of the spectrum.

Property (1) leads to a contrast increase between clouds and sky so that the clouds become more noticeable. This gives the images a more “dramatic” look. Properties (4) and (5) are mainly related to material reflectance characteristics. In the case of vegetation, chlorophyll has a very high reflection in the NIR part of the spectrum. The difference in brightness between visible and NIR images can be partly explained by the fact that blue and red radiation are absorbed to perform photosynthesis. Many vegetation-derived objects, such as textiles, share

this brightness in the NIR spectrum. The monitoring of forest growth via satellite pictures is an example of an application that uses this vegetation characteristic.

(a) Transparency of atmospheric haze

(b) Plastic transparency and vegetation brightness

(c) Textile brightness and uniformity

Figure 2.1: Some photographs of different materials. (left) RGB image, and (right) NIR image.

The NIR image properties discussed above can be divided into two main categories: scattering related (1) (2) (3) and material related (4) (5) properties. The scattering related properties can be explained by considering the Rayleigh and Mie scattering domains:

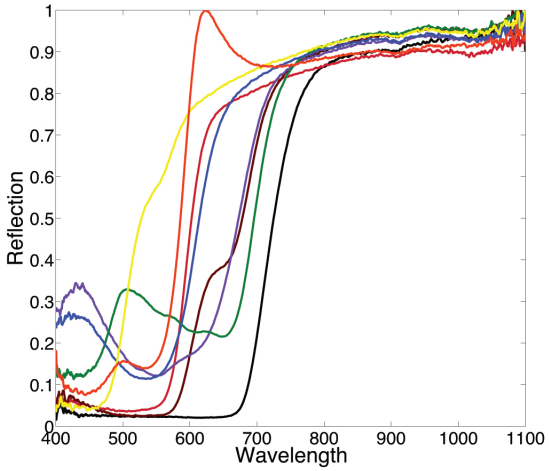


Figure 2.2: Spectral reflectance of 20 different fabrics. Their reflectances are different in the visible part of the spectrum, which for given camera and lighting conditions leads to different color values. However, as the samples belong to the same material, their spectral reflectances in the NIR range are not significantly different.

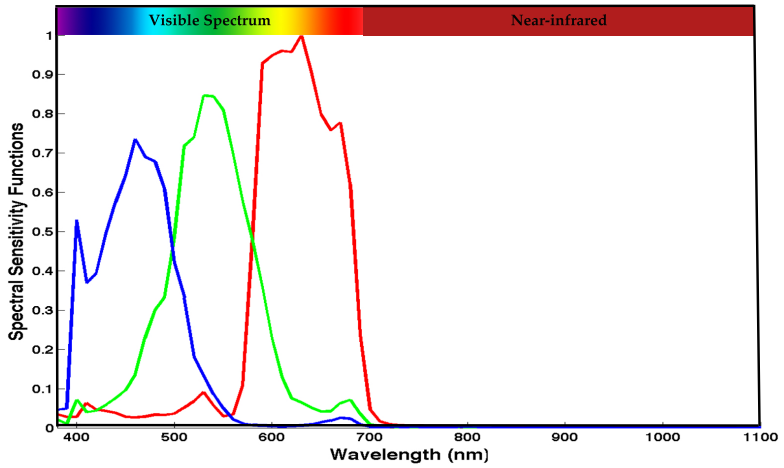
The Rayleigh equation gives the intensity of light scattered on particles with a dimension $d < \lambda/10$

$$I = I_0 \frac{1+\cos^2(\theta)}{2R^2} \left(\frac{2\pi}{\lambda}\right)^4 \left(\frac{n^2-1}{n^2+2}\right)^2 \left(\frac{d}{2}\right)^5 \tag{2.1}$$

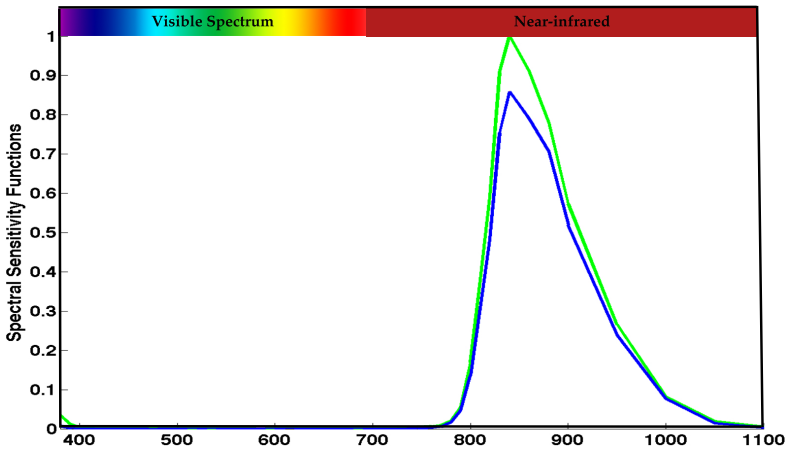
where θ is the scattering angle, n is the particle refraction index, λ is the wavelength, R is the particle distance, and d is the particle diameter. Particles in the air (haze) satisfy the size condition and are therefore subject to Rayleigh scattering.

Because of the factor $1/\lambda^4$ in Equation 2.1, the scattered intensity is higher for short wavelengths. Therefore haze scatters more light in the short-wavelength and less in the NIR range of the spectrum. Particles of clouds are larger than one tenth of the wavelength and so for them Mie scattering is dominant and the scattered intensity is no longer wavelength dependent. This is the reason cloud brightness remains almost constant in NIR.

The transmission and reflection properties of a material can change significantly according to wavelength. Therefore it is difficult to estimate the differences in brightness between visible and NIR images without having a prior knowledge of the materials in the scene. Figure 2.2 shows that for a given material, regardless of the object color in the visible part (400-700 nm), the reflection in the NIR band (700-1100 nm) remains the same.



(a) Spectral sensitivity functions with NIR blocking filter



(b) Spectral sensitivity functions with visible blocking filter

Figure 2.3: The spectral sensitivities of the NikonD90 (Figure 2.4) with B+W 486 IR/UV cut (NIR blocking filter), and B+W 093 (visible blocking filter).

2.1 How to Capture NIR Images

Current imaging sensors, both CCD and CMOS, are made of silicon and thus intrinsically sensitive to wavelengths from roughly 350 nm to 1100 nm, as illustrated in Figure 2.4. If we remove the NIR-blocking filter from the camera, the sensor has the capability of imaging both NIR and visible bands; no modification of the actual sensor is required [Fredembach and Süsstrunk, 2008].

Three main approaches have been proposed so far for RGB and NIR acquisition. In the

first approach, two sensors are placed to capture images simultaneously [Zhang et al., 2008]. Considering that the sensor is one of the expensive pieces in a camera, this approach is quite expensive. Furthermore, implementing such a system in a small device such as a cell phone is extremely difficult. The second method is to use one sensor in two consecutive shots to capture the pair of images [Fredembach and Süsstrunk, 2008]. In each shot, either a visible-blocking or NIR-blocking filter is fixed in front of the lens. This technique is time consuming and severe registration issues arise, especially in the case of dynamic scenes. The last design is to jointly capture RGB and NIR images by only a single sensor [Sadeghipoor et al., 2011]. In this method, similar to conventional color imaging, the sensor is overlaid with an array of color filters that sample the scene both spatially and spectrally. Afterwards, a reconstruction algorithm is applied to the raw image to estimate full resolution RGB and NIR image pair.

All the images captured and used in this thesis are acquired per the second method [Fredembach and Süsstrunk, 2008]. The filters used to capture visible and NIR images are B+W 486 IR/UV cut and B+W 093 Infrared filter, respectively. The spectral sensitivities of the NikonD90 (Figure 2.4) with these filters are shown in Figure 2.3.

2.2 Current Uses of NIR Imaging

Near-infrared imaging is used in different areas. NIR spectroscopy (NIRS) is employed for material identification and forgery detection. It has been shown that surface reflection in the NIR band is critical for detection of different classes of material [Burns and Ciurczak, 2001, Kulcke et al., 2003]. NIRS is a nondestructive analytical technique applied to understand the interactions between incident light and a material surface. The need for little or no preparation of samples, along with the inherent simplicity of NIRS, has made it one of the most used techniques for material identification in the industry [Burns and Ciurczak, 2001].

In remote sensing, multi-spectral images are captured to acquire information to detect, characterize, and monitor different regions (such as vegetation and soil) on the earth [Blackburn, 2007]. In such applications, region reflection in both visible and NIR parts of the spectrum is required [Zhou et al., 2009, Walter, 2004]. It has been shown that both NIR and visible wavelength ranges offer valuable information that provides bio-signatures for different classes of vegetation and soil properties [Blackburn, 2007].

In both remote sensing and NIR spectroscopy applications, hyper-spectral data is needed to accurately identify material classes. These capturing devices, however, are highly specialized and expensive, hence of limited usefulness in the ubiquitous consumer scenario we envision.

Recent studies show that combining the RGB image with even a single NIR channel (captured as mentioned above) can be successfully exploited in image processing and computer vision tasks. The use of NIR imaging combined with visible imaging has also been applied in real-time 3-dimensional depth imaging [Salvi et al., 2004]. In this application, scene depth is inferred from a pattern that is projected onto the scene. This pattern is projected using NIR illumination. Videoconferencing is another area in which NIR imaging is used [Gunawardane et al., 2010]. In this application, NIR illumination is placed on the monitor, which provides NIR radiation from several known directions. The information from the captured image in the NIR part of the spectrum is used for simultaneous relighting of the video stream. In both applications, NIR illumination is incorporated to provide more information about the scene in real-time imaging.

NIR imaging is also proposed to be incorporated into "everyday" photography. For example, Fredembach and Süsstrunk [2008] suggest a method that fuses visible images with details from NIR in landscape photography to obtain images with a greater perceived local contrast.

In face recognition tasks, NIR imagery has recently received much attention due to the high quality of acquired images as well as high performance of NIR cameras under illumination variations [Li et al., 2005, Shen et al., 2012]. It is shown that fusing NIR and visible images could improve the robustness of face recognition algorithms to illumination variations, without losing important texture information. They proposed a framework in which local binary pattern features are extracted to compensate monotonic transform and obtain illumination invariant face representation using NIR images, and statistical learning algorithms are used to decrease the dimension of features and extract most discriminative features. Guoying et al. [2011] also stated that NIR imaging is robust to illumination variations, and proposed a method that combines such images and local binary pattern features for illumination invariant facial expression recognition.

The intrinsic properties of material classes in the NIR band make this information a relevant choice in material-based segmentation and classification. Hence, we [Salamati and Süsstrunk, 2010] show that introducing NIR information into low-level segmentation makes it less probable that changes of the color within the material or changes in the lighting condition are mis-detected as object boundaries.

Closer to our work, Brown and Süsstrunk [2011] propose to use the 4-channel images (RGB+NIR) to better classify different image scenes.

As a preliminary study in this thesis, we also investigate the potential offered by NIR images to

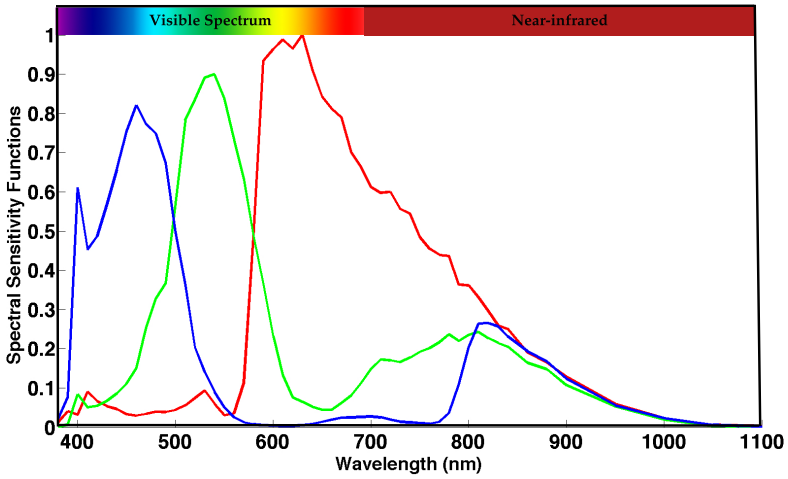


Figure 2.4: Typical transmittance curves of RGB filters of the NikonD90.

classify 4 classes of material:

2.3 Material Classification Using Color and NIR Images

In this section, we classify four different types of materials: textile, tile, wood, and linoleum. All the images were taken under controlled viewpoint and illumination conditions and their analysis was conducted in both the frequency and spatial domain. Image features include the relation between materials' intensity in the NIR and luma in the color images, texture (in the frequency domain), and color.

After extracting the relevant features and calculating the corresponding feature values, the materials were classified according to a simple probability function. The results show that our limited database is classified almost exactly, and comparisons with visible-only features show that adding NIR information yield a substantial increase in the classification rates.

2.3.1 Classification Framework

To classify the materials in the database, visible and NIR images were analyzed according to their lightness, texture, and color. The database consists of 51 wood, tile, textile and linoleum samples. The analysis results are the input to a classifier in form of feature vectors to calculate the probability of that sample to belong to a material category.

Image Analysis

The images analysis comprises three steps. First, a comparison is made between the samples' luma in the visible and NIR images, followed by a texture and color analysis.

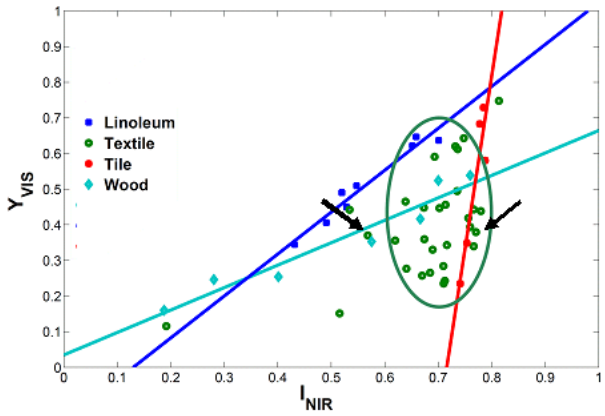


Figure 2.5: Intensity in NIR versus luma in color images. In wood, tile and linoleum, linear behavior is observed while for textile a two dimensional Gaussian can be fitted. The solid lines represent the respective linear regressions and the ellipse is the projection of the 2D Gaussian. The two textile samples specified by black arrows have roughly the same luma but different NIR intensities.

- **Luma Analysis** Since the pigments used for colorizing materials are somewhat transparent to NIR [2], we start by comparing the luma of the visible image to the NIR intensity.

  The YC_bC_r color space separates chroma information of an image from luma information. Luma (Y), which is the weighted sum of the non-linear RGB components after gamma correction, is determined by:

  $$Y = 0.2989R' + 0.5870G' + 0.1140B' \tag{2.2}$$

  Where R', G' and B' are the normalized $sRGB$ values.

  Y in color images (Y_{VIS}) and intensity in NIR images (I_{NIR}) [Fredembach and Süsstrunk, 2008] are calculated for all samples. Figure 2.5 plots I_{NIR} versus Y_{VIS} for all samples in the database. Note that the intensity in the NIR images is always higher than luma in the visible images. In addition, samples in each of the wood, tile, and linoleum classes form a line, which can be represented by a linear regression, The regression lines show high correlation coefficient values of $r^2 = 0.93$, 0.86, and 0.93, for tile, wood, and linoleum, respectively.

The I_{NIR} for almost all textile samples, however, lies in the narrow range between 0.6 and 0.8. Y_{VIS}, on the other hand, lies in a broader range. As illustrated in Figure 2.5, the relation between textile intensity in NIR and luma in color images can be modeled by a two dimensional Gaussian, whose mean and variance are:

$$\mu_v = 0.4 \quad \mu_h = 0.7 \;\; \sigma_v = 0.13 \quad \sigma_h = 0.07 \tag{2.3}$$

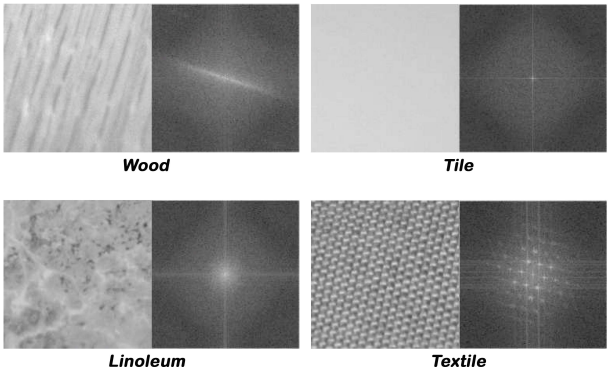


Figure 2.6: Samples in the spatial (left) and frequency (right) domain. The frequency spectra shows energy patterns that are characteristic of the materials' surfaces. A line can be observed in the wood sample due to the existing parallel lines on the surface. The peaks for the textile sample are due to the nature of the woven fabric. In tile and linoleum there exists high energy in low frequencies.

- **Texture Analysis** Images of real objects often do not exhibit regions of uniform lightness. For instance, the image of a wooden surface contains variations in intensities that form certain repeated patterns due to its specific surface characteristics. Figure 2.6 displays samples from different materials in the spatial and corresponding frequency domain. Due to the nature of woven fabric, where the surface is formed by a series of straight parallel lines crossing each other, a number of peaks in certain frequency bands can be witnessed. The naturally existing parallel lines on the wood surface result in the formation of a line at a specific angle in the frequency domain. Tiles' smooth surface leads to most energy being located in low frequencies.

  To analyze these texture characteristics, we use here filters adapted from [Randen and Husoy, 1999], namely ring and rectangular filters. Ring filters are used for analyzing the energy in certain frequency bands, while rectangular filters are employed for orientation

detection. Ring filters ${H_{ring}}^{(i)}$ are Gaussian functions and defined according to:

$$ {H_{ring}}^{(i)}(w_1, w_2) = \exp\left(-\left(\frac{\left(r-\mu_i\right)}{\sigma_i}\right)^2\right) \tag{2.4} $$

$$ r = \sqrt{w_1^2 + w_2^2} \tag{2.5} $$

where w_1, w_2 are spatial frequencies in the spatial domain. μ_i and σ_i are parameters determining the center frequency and the bandwidth of each ring filter, respectively. The filters are nondirectional (See Figure 2.7). In this work, 13 ring filters have empirically been chosen and applied.

The other set of filters are rectangular filters H_{rect} that are constructed according to:

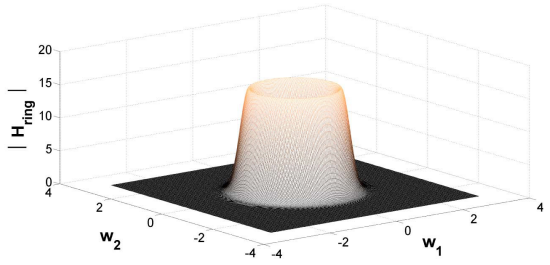
$$ w_1 - \tan(\theta \times w_2) - \frac{0.1}{2 \times \cos\theta} \leq H_{rect} \leq w_1 - \tan(\theta \times w_2) + \frac{0.1}{2 \times \cos\theta} \tag{2.6} $$

where θ varies from 0 to π, i.e., the rectangular filter is rotated and the energy is calculated at each angle. The width of the filters is constant and empirically chosen to be 0.2.

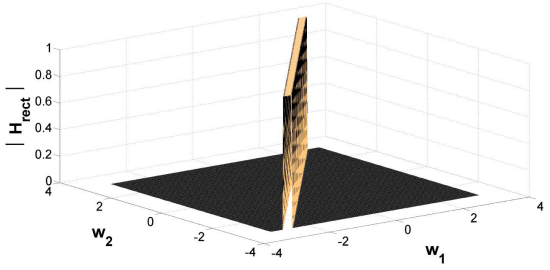
Different materials can be colorized in such way that their patterns give the same features as other materials in the texture analysis. Knowing that some colorants used in printed material are usually transparent to NIR, using NIR images for texture analysis avoids such problems.

- **Color Analysis** The process of colorizing a manufactured object is complex and varies according to the material; the colorants themselves are also diverse. Although different colorants may look identical in color images, they have different responses in NIR images. Therefore, color information of the samples and the corresponding NIR intensities can be an important cue.

  We employ the hue, saturation and luminance (HSL) space, used in color image processing. Although almost all colors of the visible spectrum can be produced by merging primaries, the process of colorizing different material makes each class of material capable of having only a limited gamut (i.e., the set of possible colors within a material) of the visible spectrum (see Figure 2.8 for illustration). Samples that have the same luma in color images may or may not have the same NIR intensity (see Figure 2.5). The



(a)



(b)

Figure 2.7: Representations of one of the a) ring and b) rectangular filters in the frequency domain. The height of the surface above the w_1, w_2 plane and the color level values represent the filter's amplitude.

two textile samples indicated by black arrows in Figure 2.5 have roughly the same luma but different NIR intensities. The hue values in these two samples are different (see Figure 6.6), i.e., the difference in NIR intensity for the same material can be related to the difference in hue. Hence, analyzing the gamut of each existing material, obtained using hue and saturation from the color image and the intensity from the NIR image of each sample may lead us to a better classification of material.

To do so, the RGB values of each sample are first converted into HSL and the luminance is replaced by the intensity of the NIR image. A 3-D convex hull algorithm was applied to determine the position and the volume of the gamut for each material class.

From Features to Classification

In this section, we explain how to obtain feature vectors and classify materials from the acquired features. First, features are selected and then the probability of a sample to belong to a material class is calculated. We explain how to calculate the probability pertinent to each analysis.

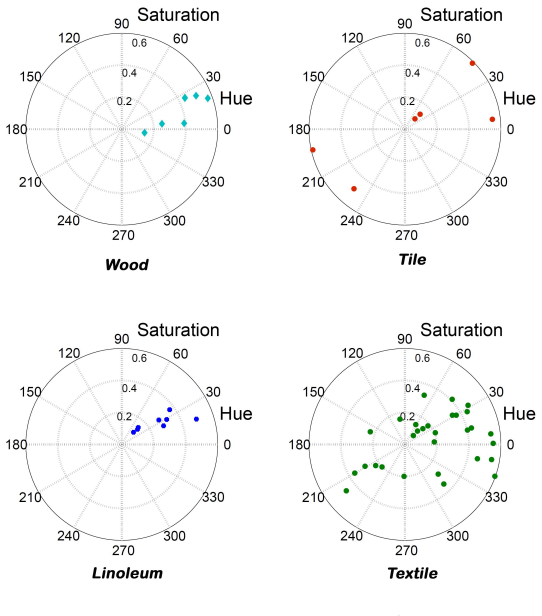
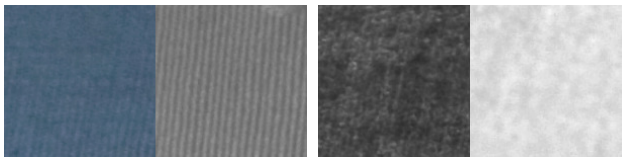


Figure 2.8: Hue versus saturation in color images. In wood samples, hue varies within a narrow angle, however, they have a wide range of saturation. To be expected tile and textile samples cover almost all hue and saturation values. The linoleum samples in our database are also within a narrow hue range and are not very saturated. However, this is due to our limited sample selection.



(a) (b)

Figure 2.9: Images of two textile samples with the same luma in the visible part of the spectrum but different NIR intensity. The difference in intensity in the NIR images can be related to the difference in their hue.

- **Feature Extraction** From each analysis, features are selected to quantify material characteristics. The choice of appropriate descriptive parameters will significantly influence the effectiveness of the classification.

  Intensity in NIR and luma in color images are taken to be feature values T_1 and T_2.

  $$T_1 = I_{NIR} \qquad T_2 = Y_{VIS} \tag{2.7}$$

  For texture, we normalized the energy existing in different ring filters.

  $$\bar{E}^{(i)}_{ring} = \frac{E^{(i)}_{ring}}{\max_{i=1\ldots13}\left(E^{(i)}_{ring}\right)} \tag{2.8}$$

  $$E^{(i)}_{ring} = \|F_I(w_1, w_2) \times H^{(i)}_{ring}\|^2 \tag{2.9}$$

  Where $\bar{E}_{ring}$ is the normalized energy, $F_I(w_1, w_2)$ is the Fourier transform of the image I, and $\|.\|$ denotes the Frobenius norm (see Fig. 2.10 for a representative sample from each material category).

  For all the non-textile samples, the larger the diameter of the ring filter, the less the spectral energy in the corresponding ring filtered image, i.e., most of the energy lies in the low frequency part of the spectrum. For the textile samples, however, the existing peaks in the high frequencies due to the nature of the woven fabric will result in an increase of the energy in the ring filtered images incorporating those peaks. Hence, the filtered image for which the energy is maximum can be taken into consideration as a feature value, which we call T_3. For simplicity, this feature value can be reduced to a

binary value:

$$T_3 = \begin{cases} 0 & \text{if } argmax_{i=1...13}\left(\bar{E}_{ring}^{(i)}\right) = 1 \\ 1 & \text{otherwise} \end{cases} \tag{2.10}$$

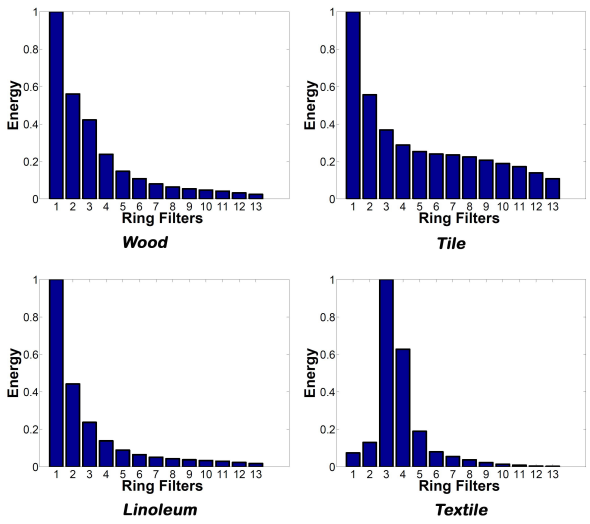


Figure 2.10: Relative energy $\bar{E}_{ring}^{(i)}$ in all 13 ring filtered images for a random sample in each class. The corresponding T_3 value for each sample is: [0,0,0,1].

Figure 2.11 displays the energy existing in different rectangular filters for all angles between 0 and π, one degree interval, for a random sample from each material category. In the smooth samples, like tiles, the energy is constant at all angles, for oriented texture samples (such as wood), an energy peak is observed at a specific angle. For textile samples, due to the existence of peaks in the frequency domain, more than one peak of energy is observed at some specific angles.

The energy peak at a certain angle is detected when its distance to the energy of surrounding angles is more than a certain threshold. The considered threshold δ will reduce the sensitivity of the algorithm to noise, thus it is taken as the variance of that signal,

$$\delta_x = \frac{\sum_{i=1}^{180}\left(E_x{}^{(i)} - \bar{E}_x{}^{(i)}\right)^2}{180} \tag{2.11}$$

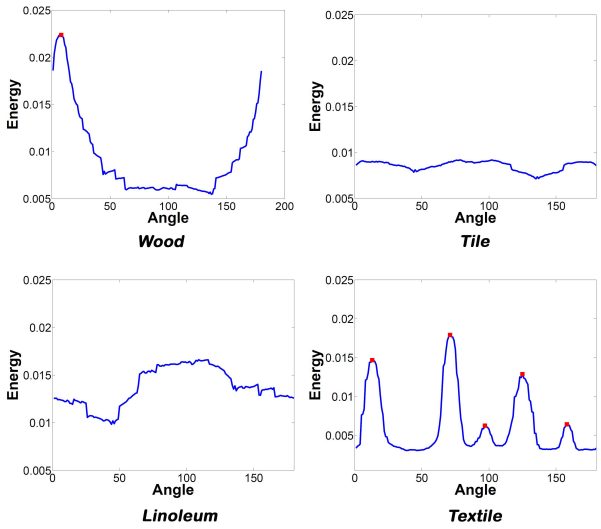


Figure 2.11: The energy in the rectangle filter from 0 to 180 degrees for a random sample in each class. The detected peaks are marked by red circles. The corresponding T_4 value for each sample is: [1,0,0, 2].

$$E_x^{(i)} = \|F_I(w_1, w_2) \times H_{rect}^{(i)}\|^2 \tag{2.12}$$

where ${E_x}^{(i)}$ is the energy existing in the i^{th} rectangular filtered image and $\bar{E_x}^{(i)}$ is the average of the energy over all rectangular filters. As a result, the existence of one or more peaks at an angle is used as the feature value T_4.

$$T_4 = \begin{cases} 0 & \text{if } N_P = 0 \\ 1 & \text{if } N_P = 1 \\ 2 & otherwise \end{cases} \tag{2.13}$$

Where N_p is the number of detected peaks.

The fifth feature vector contains the hue, saturation, and NIR intensity coordinates.

$$T_5 = \{Hue, Saturation, I_{NIR}\} \tag{2.14}$$

- **Luma Related Probability** As seen in section 2, wood, tile, and linoleum have Y_{NIR} varying linearly with Y_{VIS}. This behavior is easily modeled by linear regression:

$$\hat{Y_{ij}} = X_{ij}\hat{\beta}_i \tag{2.15}$$

where $\hat{\beta}_i = \left({X_i}^T X_i\right)^{-1} {X_i}^T \hat{Y}_i$, and $\hat{Y}_i = X_i \left({X_i}^T X_i\right)^{-1} {X_i}^T Y_i$. Y_i is the luma of the visible images and X_i contains the corresponding intensity in NIR images.

From this, the vector of residuals can be defined as [Cook and Weisberg, 1982]:

$$\hat{\epsilon_i} = Y_i - \hat{Y}_i \tag{2.16}$$

By studentizing the residuals [12], we are able to determine the probability of sample x belonging to tile, wood, or linoleum. For each line, the variance of the residuals for all samples can be calculated under a certain confidence level α_i. Variance of residual of each sample defines the distance from the regression line that shows how much further from the regression line a new samples in that material class can fall. In other word, we can be α_i% sure that all samples in the material class fall in the area around the regression line that has been defined by the maximum variance of residual. Moreover, we are $(1-\alpha_i)$% sure that our target sample belongs to the line representing the class when that sample is located outside of that area. The probability for any new sample to belong to each class can be calculated by the maximum confidence interval that makes an area so that the target sample is outside of the area:

$$P_L\left((x \in A_{i=1\cdots3}) \mid T_1, T_2\right) = 1 - \alpha \tag{2.17}$$

where α is the maximum confidence interval forming an area to which sample x doesn't belong, $A = \{A_i \mid i = 1\cdots3\}$ represent tile, wood, and linoleum, respectively. The residual variance is calculated within the confidence interval of α, thus $1-\alpha$ is the probability that the sample belongs to that class (see Fig. 2.11 for illustration).

For the textile class, we model the relationship between I_{NIR} and Y_{VIS} as a 2D Gaussian function. $P(textile \mid T_1, T_2)$ is thus given by:

$$P_L\left((x \in A_4) \mid T_1, T_2\right) = \frac{1}{2\pi\sigma_v\sigma_h} e^{\frac{-1}{2}\left(\frac{T_1^2-\mu_v}{\sigma_v} + \frac{T_2^2-\mu_h}{\sigma_h}\right)} \tag{2.18}$$

where σ_v and σ_h are the variance and μ_v and μ_h are the mean of the two Gaussians

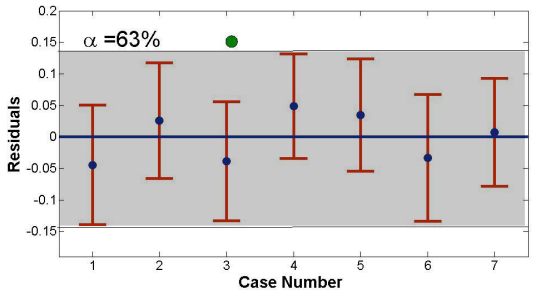


Figure 2.12: The residuals of the wood samples within 63% confidence interval. The black line is the regression line representing the correlation of the wood samples in the database. The area in which we are 63% confident that samples are wood is shaded gray. Thus the probability of the target sample (green point) to be wood is 37%.

with respect to the color and NIR images. The values of σ and μ were given in section 2. We are, however, interested in the probability of a sample belonging to one class (A_i) but not the other $\left(x \neq A_{j\neq i}\right)$:

$$P_L\left(\left(x \in A_i \cap \left(x \neq A_{j\neq i}\right)\right) \mid T_1, T_2\right) = \frac{P\left(x \in A_i \mid T_1, T_2\right) \times P\left(A_i\right)}{\sum_{i=1}^{n}\left(P\left(T_1, T_2 \mid A_i\right) \times P\left(A_i\right)\right)} \tag{2.19}$$

where

$$P\left(A_i\right) = \frac{N_i}{N_T} \tag{2.20}$$

and N_i is the number of samples existing in the i^{th} material in the database and N_T is the total number of samples in the database.

Texture Related Probability

In order to calculate the probability of a sample to belong to a material category, knowing T_3 , T_4, the Bayes theorem is used.

$$P_T\left(\left(x \in A_i\right) \mid T_3, T_4\right) = \frac{P\left(T_3, T_4 \mid A_i\right) \times P\left(A_i\right)}{\sum_{i=1}^{n}\left(P\left(T_3, T_4 \mid A_i\right) \times P\left(A_i\right)\right)} \tag{2.21}$$

$P\left(T_3, T_4 \mid A_i\right)$ can be defined as number of cases favorable for the feature vector $[T_3, T_4]$, over the number of total samples in material A_i. We calculate this probability for each feature vector, i.e., the probability of each sample of a certain material to have a certain feature value $[T_3, T_4]$ (see Table 2.1).

- **Color Related Probability** Knowing the position of a new sample in H, S and I_{NIR} color space (T_5) as well as the gamut for each material class, we can conclude that if (T_5) exists

T_3	T_4	**Textile**	**Tile**	**Wood**	**Linoleum**
1	0	4/30	0	0	0
1	1	6/30	0	0	0
1	2	13/30	0	0	0
0	0	3/30	6/6	7/8	0
0	1	4/30	0	1/8	7/7
0	2	0	0	0	0

Table 2.1: $P(T_3, T_4 \mid A_i)$ in the database.

in the gamut of material A_i, then that sample will belong to the class A_i. Therefore, the following statement can be used to specify the probability of a sample to belong to the material class A_i:

$$P_C(x \in A_i) = \begin{cases} \frac{1}{n} & \text{if } T_5 \in Gamut_{A_i} \\ 0 & \text{if } T_5 \notin Gamut_{A_i} \end{cases} \tag{2.22}$$

where n is the number of material classes whose gamuts intersect at the position of T_5.

Final Probability Estimation

We assume that the probabilities resulting from luma and color analysis is independent from texture analysis, so for that neither luma nor color of a material impact the surface characteristics of that material.

To investigate the dependency of color and luma analysis, we should mention the fact that three attributes of color in HSL are decorrelated [13], i.e., knowing the relation between luma and NIR intensity does not give us any information about the relation between hue, saturation and NIR intensity.

Thus the corresponding probabilities are independent. The final probability for each sample can be calculated by multiplying the probabilities given from each analysis:

$$P(x \in A_i \mid T_1, \cdots, T_5) = P_L \cap P_T \cap P_C = P_L \times P_T \times P_C \tag{2.23}$$

$P(x \in A_i \mid T_1, \cdots, T_5)$ represents the probability of a sample x to belong to a material category A_i according to feature values T_1 to T_5.

	Textile	**Tile**	**Linoleum**	**Wood**
Textile	25/30	0	0	5/30
Tile	0	6/6	0	0
Linoleum	0	7/8	0	1/8
Wood	0	0	1/7	6/7

Table 2.2: The confusion matrix using just visible information

	Textile	**Tile**	**Linoleum**	**Wood**
Textile	29/30	0	0	1/30
Tile	0	6/6	0	0
Linoleum	0	0	8/8	0
Wood	0	0	0	7/7

Table 2.3: The confusion matrix using both visible and NIR information.

2.3.2 Experiment

All samples were photographed in the visible and in the near-infrared range of the spectrum in a controlled environment. The camera we used in these experiments is a Canon EOS 300D and the light source was incandescent. The photography operation followed the same procedures for the two types of photographs taken from the samples. The database on which the training and testing were conducted consists of 30 textile, 5 tile, 8 linoleum and 7 wood samples.

In order to determine how accurately this learning algorithm will be able to predict a new sample's material, leave-one-out cross validation has been applied. When using the leave-one-out method, the learning algorithm is trained multiple times, using all but one of the data points and then testing the removed data point and calculating the probability of that sample to belong to each class.

For the entire database, the feature vectors are formed and the probability of each sample having a certain feature vector, given the material, is calculated. The probability of each sample belonging to each material given the feature vector is calculated according to Equation 2.23 for each left-out sample.

The algorithm was applied to all the samples in our dataset and the probability of each left-out sample belonging to each material category was calculated. The higher the probability, the better we could come to the conclusion that the sample belongs to a certain category.

To assess the usefulness of NIR information, the classification was performed using visible

features only (results in Table 2.2) as well as using visible and NIR features (results in Table 2.3). We see that the additional NIR information makes the proposed classification more accurate. The wood and textile samples were classified better due to transparency of most of the colorants to the NIR light in the sample; as a result the NIR images provide more effective data.

2.4 Conclusion

In this chapter, we have discussed NIR imaging and particularities of such images, as well as common difficulties of capturing in this part of the spectrum using a normal digital camera. We have also reviewed the relevant literature on incorporating NIR into digital photography and computer vision tasks. We showed that the relation between the visible and NIR information yields an improvement in classification of 4 different types of material: textile, tile, wood, and linoleum. The material classes were more accurately classified when NIR information was present. Whereas the result and framework are suited for the samples in our database and they may not generalize to more material classes, one could still conclude that incorporating the intrinsic characteristics of NIR images leads to an improvement of the accuracy of image classification tasks, which we will show in later chapters.

3 Tools Used in the Thesis

My research for this thesis draws on prior work in different aspects of image classification and segmentation in scene understanding.

In the computer vision community, most of the earlier object and scene recognition work assigns a single label to an image, for example an image of a forest, a mountain, or a beach. Some go further by creating a list of annotations without localizing where in the image each annotation belongs. Previous work on low-level image segmentation mostly considered local descriptors and often ignored the photo contents. More recent work, however, has discovered that visual matching and modeling of object appearance can be of great assistance. In this chapter we discuss image representations and machine learning techniques for image classification and semantic segmentation. A short background is also provided on low-level boundary detection. Another area of relevance includes forming intrinsic images in order to find image boundaries that are robust to different illumination conditions.

This chapter provides, for the convenience of a reader unfamiliar with them, the image representation and machine learning concepts used in the later chapters, and helps placing my contribution in the wider context of the state of the art in these fields.

3.1 Image Representations for Classification

In this section, we discuss image classification frameworks with a special emphasis on the Fisher Vector (FV) representation [Perronnin and Dance, 2007, Perronnin et al., 2010]. We focus on this representation and also give a short overview of Bag-of-Visual-Words (BOW) [Sivic and Zisserman, 2003, Csurka et al., 2004], as they are efficient and have shown to be among the state-of-the-art representations used for image classification.

For any image processing operation, we need to represent an image by features extracted therefrom. The image is usually represented by a set of local feature descriptors extracted from a set of regions in the image. There are generally two attributes for local features: a feature detector and a feature descriptor [Forsyth and Ponce, 2002, Mikolajczyk and Schmid, 2004]. A feature detector detects interesting locations in the image, for example corners and edges. A feature descriptor describes the image patch around that interest point, usually by histograms of gradients or orientation.

- **Feature Detectors:**

  The first approach in detecting features is based on key-points detection, where image content is used to select a set of points that are of specific interest. There are different kinds of feature detectors, but among the most common ones are difference of Gaussians (DoG) [Lowe, 2004], Hessian affine, and Harris affine [Mikolajczyk and Schmid, 2004].

  The second approach, that we also use it in this thesis, is based on dense sampling of points from the image [J. Winn and Minka, 2005]. All points on a regular dense grid over different scales are used as key-points. The main reason to use dense sampling is to avoid early removal of potentially interesting points.

- **Feature Descriptors:**

  Feature descriptors are used for describing local image features. The aim of a descriptor is to find an image feature and describe it in a way that is not affected by perspective, scale, occlusion or illumination. One of the most common methods for this is scale-invariant feature transformation (SIFT), that was developed by Lowe [1999], this is considered to be one of the most robust feature descriptors [Bauer et al., 2007]. The speeded-up robust features (SURF) descriptor developed by Bay et al. [2006] is a method inspired by SIFT and it is considered to be equally robust as SIFT and is more efficient.

  The SIFT descriptor describes a sampled point, by a histogram of image gradients (illustration in Figure 3.1). The gradients are computed over the intensity levels of the image, and aggregated in several spatial bins around the sampled point, using both the magnitude and the orientation. As a result of using gradients, the descriptor is invariant to the intensity changes. Also, due to the use of spatial histograming of gradients, the descriptor is to some extent robust to geometric distortion.

  To increase discriminative power, color descriptors have also been used [Clinchant et al., 2007, Perronnin et al., 2010]. Local RGB color statistics computes the mean value and standard deviation of each RGB channel around the sampled point using a 4x4 spatial

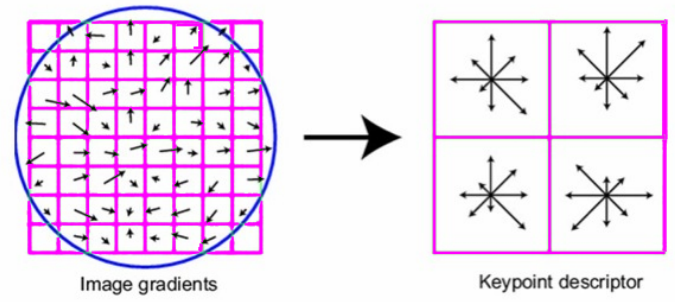


Figure 3.1: Illustration of the SIFT descriptor, Around each keypoint, a 8 × 8 window is formed and divided into 4 × 4 cells. Within the window, gradient magnitude are computed. For each cell, accumulate an 8-orientation histogram, then concatenate them to form a 4 x 4 x 8 = 128-dimensional vector. Image courtesy of Lowe [1999].

grid, like the SIFT descriptor. This feature possesses no invariance properties, and is often combined with invariant descriptors such as SIFT.

The features that we use in this thesis are SIFT, color, and the late fusion of SIFT and color.

The notion of image classification in the user's mind is mostly based on high-level concepts (or semantics), such as activities, objects, or events. Therefore, classification by similarity using low-level features like color or texture will not be very effective. Hence, an intermediary level representation is introduced as a first step between low-level descriptors and scene classification in order to deal with the semantic gap between low-level features and high-level concepts. In the image classification literature, the traditional approach to transform low-level features into high-level representations is the bag-of-visual-words (BOW) [Sivic and Zisserman, 2003, Csurka et al., 2004]. Recently, Perronnin and Dance [2007] proposed Fisher representation as an alternative to BOW at the patch level.

3.1.1 Bag-of-Visual-Words

The BOW methodology was first proposed in the text retrieval domain problem for text document analysis, and it was further adapted for computer vision applications [Sivic and Zisserman, 2003, Csurka et al., 2004]. For image analysis, after feature detection and description, a visual vocabulary is formed in the BOW model, which is based on the vector quantization process by clustering low-level feature descriptors of local regions, such as color, texture, or a combination of them. Figure 3.2 describes these four steps to extract the BOW features from images.

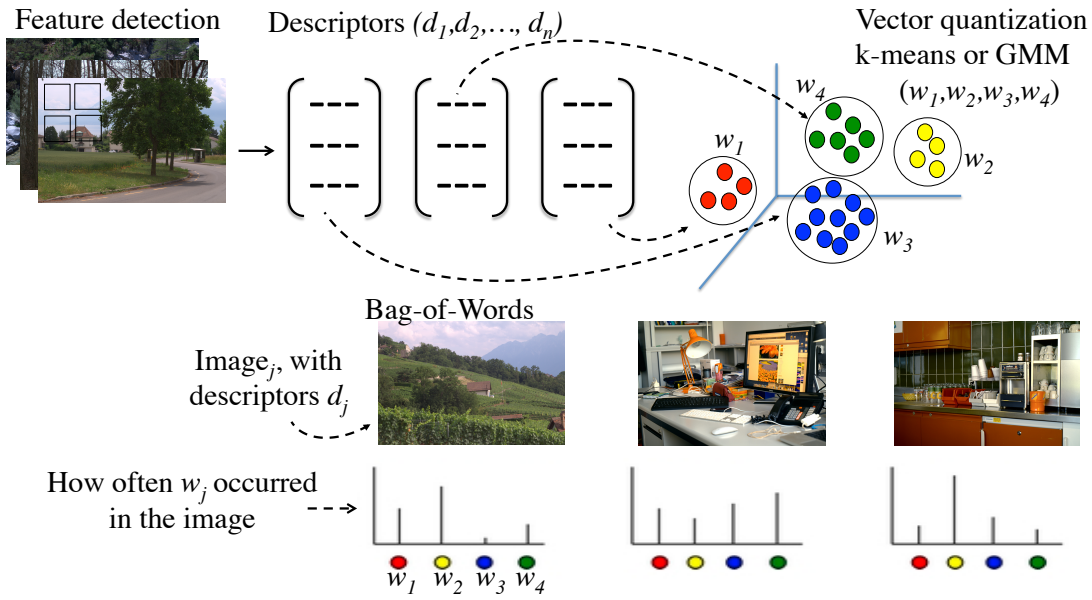


Figure 3.2: Different steps for constructing the bag-of-words for image representation. Image adapted from [Sun et al., 2010].

Given a training dataset containing n images, $d_1, d_2, \cdots, d_n$ are extracted local features for each image. The visual dictionary (codebook) is often created from this large set of local image descriptors. A specific unsupervised learning algorithm, such as k-means or Gaussian Mixture Models (GMM), is used to group the training set into a fixed number $|W|$ of visual words $W = w_1, w_2, \cdots, w_{|W|}$. Experimentally it has been shown that classification performance improves with the size of the dictionary [van Gemert et al., 2010, Chatfield et al., 2011]. Examples of other approaches to learn a codebook are: mean shift clustering to create a codebook that better represents a non-uniform distribution of descriptors [Jurie and Triggs, 2005], or sparse dictionary learning that learns a codebook to minimize the reconstruction error of local descriptors [Mairal et al., 2008, Wang et al., 2010]. Each of the local features can now be encoded using the learned codebook. The goal is to represent the original local feature by one or more visual words such that a reconstruction error or expected distortion function is minimized. The most frequently used encoding is probably vector quantization (VQ, also known as hard-assignment): a local feature is assigned to its nearest neighbor in the dictionary. To reduce the reconstruction error, local features can be encoded using soft-assignment. If the codebook is based on a mixture of Gaussians, the posterior probabilities of each Gaussian can be used as weights in the soft-assignment. As the final step, we can summarize the data in a $|W| \times n$ concurrence table of counts $N_{ij} = n(w_i, d_j)$, where $n(w_i, d_j)$ denotes how often the word w_i occurred in an image with descriptor d_j.

3.1.2 Fisher Vectors

The Fisher Vector (FV) [Perronnin and Dance, 2007] is an extension of the BOW representation [Csurka et al., 2004], which, instead of describing images as histograms of visual word occurrences, considers higher order statistics [Jaakkola and Haussler, 1999, Perronnin and Dance, 2007].

The principle is the following: First, a set of low-level features is extracted from each image. Patches are extracted according to a regular grid (dense detector) and a local descriptor is computed for each patch. As the next step, a principle component analysis (PCA) projection is applied to the descriptors. Projected descriptors are used to build a visual codebook that describes the descriptor space as a Gaussian mixture model (GMM), with N Gaussians. The GMM distribution can be written as:

$$u_\lambda(x) = \sum_{i=1}^{N} w_i \mathcal{N}(x|\mu_i, \Sigma_i) \tag{3.1}$$

This visual codebook is used to transform each image into a global signature (FV). To compute the FV for one image, we consider its set of low-level features $X = \{x_t, t = 1 \dots T\}$. The FV $\mathcal{G}_\lambda^X$ characterises the sample X by its deviation from the distribution u_λ (with parameters $\lambda = \{\mu_i, \Sigma_i, i = 1..N\}$):

$$\mathcal{G}_\lambda^X = L_\lambda G_\lambda^X \tag{3.2}$$

where G_λ^X is the gradient of the log-likelihood with respect to λ:

$$G_\lambda^X = \frac{1}{T} \nabla_\lambda \log u_\lambda(X) \tag{3.3}$$

L_λ is the Cholesky decomposition of the inverse of the Fisher information matrix F_λ of u_λ, i.e., $F_\lambda^{-1} = L'_\lambda L_\lambda$, where by definition:

$$F_\lambda = E_{x \sim u_\lambda} \left[\nabla_\lambda \log u_\lambda(x) \nabla_\lambda \log u_\lambda(x)' \right] \tag{3.4}$$

More details on the FV computation and its theoretical foundations can be found in Jaakkola and Haussler [1999], Perronnin and Dance [2007]. As suggested in Perronnin et al. [2010], square-root and L2-normalisation needs to further be applied to the FV based image signature. Due to its high performance, we use FV as an image representation for our study.

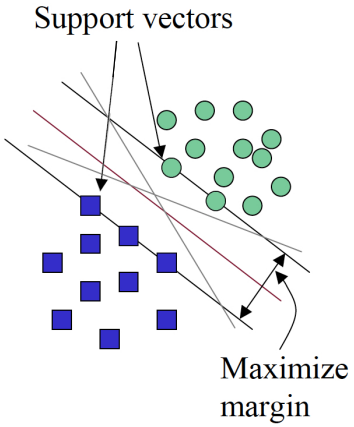


Figure 3.3: SVMs maximize the margin around the separating hyperplane.

3.1.3 Learning Model

After the BOW or FV feature is extracted from images, it is entered into a classifier for training or testing. The learning task is to compute a classifier or model $\hat{f}$ that approximates the mapping between the input-output examples and correctly labels the training set with some level of accuracy. This can be called the training or model generation stage. After the model $\hat{f}$ is generated or trained, it is able to classify an unknown instance into one of the learned class labels in the training set.

Most of the widely developed classifiers are based on support vector machines (SVMs). SVMs are designed to solve two-class problems[1]. Two approaches can be used for a multi-class problem:

1. One against all: $|C|$ classifiers are iteratively applied on each class against the rest, the highest scoring label is kept for each vector.
2. One against one: $\frac{|C|(|C|-1)}{2}$ classifiers are applied on each pair of classes, the most often computed label is kept for each vector.

The aim of SVMs [Vapnik, 1999] is to find the separating hyperplane, to which the distance to the nearest training data points on each side is maximal. These are called support vectors and their distance is the optimal margin (see Figure 3.3).

SVMs in their base form assign one of two labels to each data point in feature space by dividing

[1] We only consider linear, single-class SVMs here, as they work well with the Fisher Vectors we use as feature representation.

it into two half-spaces, one for each label. In particular, they divide it along the hyperplane that has the largest distance to the nearest training data points of both classes. The key insight is that a larger margin equals better generalization performance, and hence, a higher performance on the test case. Thus, SVMs consider the loss function

$$l(\mathbf{x}^i, y_i, \mathbf{w}) = \max(0, 1 - y_i \mathbf{w}^\top \mathbf{x}^i) \tag{3.5}$$

and minmize $\sum_i l(\mathbf{x}^i, y_i, \mathbf{w})$, where the $\mathbf{x}^i$ are the training feature vectors, while $\mathbf{w}$ and the y_i are the parameters to be varied. The offset 1 enforces the margin, since zero loss implies $y_i \mathbf{w}^\top \mathbf{x}^i \geq 1$.

While a basic SVM outputs only the classification $\text{sign}(\mathbf{w}^\top \mathbf{x})$, the method known as Platt scaling [Platt, 1999] is frequently used to obtain a probabilistic output.

3.2 Boundary Detection and Low-Level Segmentation

Boundary detection is a low-level operation that aims at partitioning images by determining homogeneous regions and forming a boundary around them. This task is important in several applications of image processing and computer vision since it represents the very first step of low-level processing of imagery.

This section provides a review of the method we used for the low-level segmentation in Chapter 6.

3.2.1 Mean Shift

Mean shift is a clustering algorithm that has been published in the context of image segmentation [Comaniciu and Meer, 2002]. The main idea behind the algorithm is to compute for every single pixel a series of mean values in the feature space. The mean is shifted towards more densely populated regions in the feature space.

The Algorithm

Each pixel is represented as a point x_i in a feature space. In their article, the authors propose to use a 5-dimensional feature space with two axes for the spatial position and three axes for the CIELUV values of every pixel. Then, for every single pixel x_i, the same procedure is undertaken:

1. Initialize $y_{i,1} = x_i$ as a starting point

2. Calculate a series of shifted means

$$y_{i,j+1} = \sum_{i=1}^{n} x_i \frac{K}{h_s^2 h_r^2} G\left(\left|\left| \frac{y_{i,j}^s - x_i^s}{h_s} \right|\right|^2\right) \cdot G\left(\left|\left| \frac{y_{i,j}^r - x_i^r}{h_r} \right|\right|^2\right) \quad j = 1,2,\ldots$$

   where the superscripts s and r denote the spatial and the range part of the feature vector respectively. The function G is usually a Gaussian. The constant K is a normalizing factor. h_s and h_r are tunable parameters, see below for their interpretation.

   The series ends when it converges after m iterations. The convergence point is called $z_i = y_{i,m}$.

3. Delineate in the joint domain the clusters C_j, $j = 1 \ldots k$ by grouping together all z_i that are closer than h_s in the spatial domain and h_r in the range domain.

4. Each pixel x_i is assigned the label C_j with $z_i \in C_j$.

5. Refinement step: Eliminate labels that occur on less than a threshold number of pixels.

There is a very intuitive and easy to understand explanation for this mean shifting. At every step, it shifts the mean towards the highest density in the local neighborhood of the joint spatial-range domain. The convergence point is the point of maximal point density. This process is illustrated in Figure 3.4. The choice of the two parameters h_s and h_r influences the granularity of the segmented result. The higher h_s and h_r, the fewer classes the algorithm will create.

In the work by Christoudias et al. [2002], the mean shift idea of Comaniciu and Meer [2002] is combined with an edge-detection algorithm, resulting in a synergetic method that exploits both algorithms' strengths. This algorithm is implemented in EDISON[2], a software that allows various combinations for test purposes.

Results

Figure 3.5 shows a Baboon image and its filtered versions for different parameters h_s, h_r. When looking at the first column and the first row, the filtering effects for different parameter combinations can be observed. Two regions show the effects well: the beard and the reflections on the cheeks. An increasing spatial parameter h_s (first column) causes the reflections to

[2] EDISON: **E**dge **D**etection and **I**mage **S**egmentati**ON**

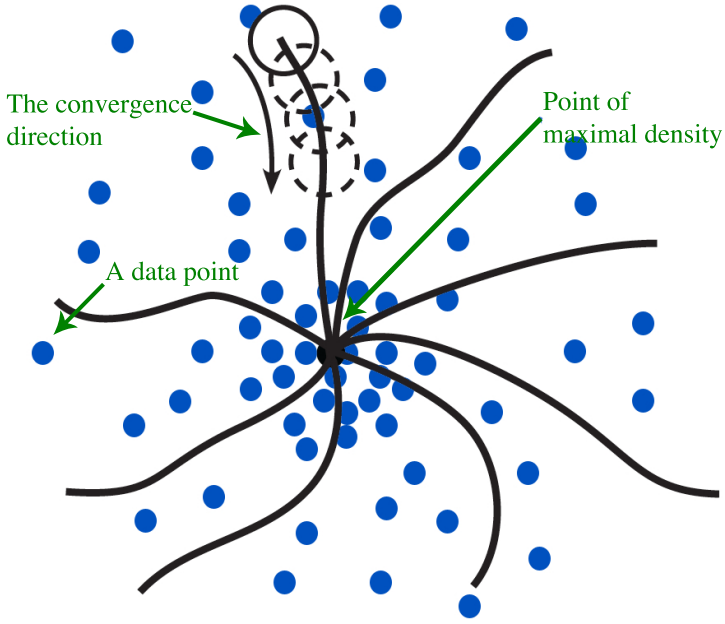


Figure 3.4: Illustration for mean shift algorithm.

be more and more blurred. However, the beard does not loose too much contrast, since the range in value (h_r) is rather small. In the case of increasing range value (first row) the beard gets blurred, however, the reflections on the cheeks remain. This is because the beard has high-value changes on a small spatial range and the cheeks have a low-value change on a large spatial range.

Discussion

A very useful property of this algorithm is that it can separate two interlaced classes in feature space as shown in Figure 3.6. If most of the points are within the regions and get denser towards the middle, the mean will be shifted along the contour of each region. It is evident that this is only true if the parameters h_s, h_r are not too large and thus the neighborhood for the mean calculation is not much bigger than the width of each region.

The problem is that the choice of the parameters h_r, h_r is not automatic. The extreme cases, where they tend to zero or infinity, can be excluded as not useful, but a proper choice is not evident. Basically, smaller values will lead to more classes since the mean shifting procedure has a smaller reach in the feature space.

For an arbitrary image, the number of classes is not a priori clear. Hence, one strength of this

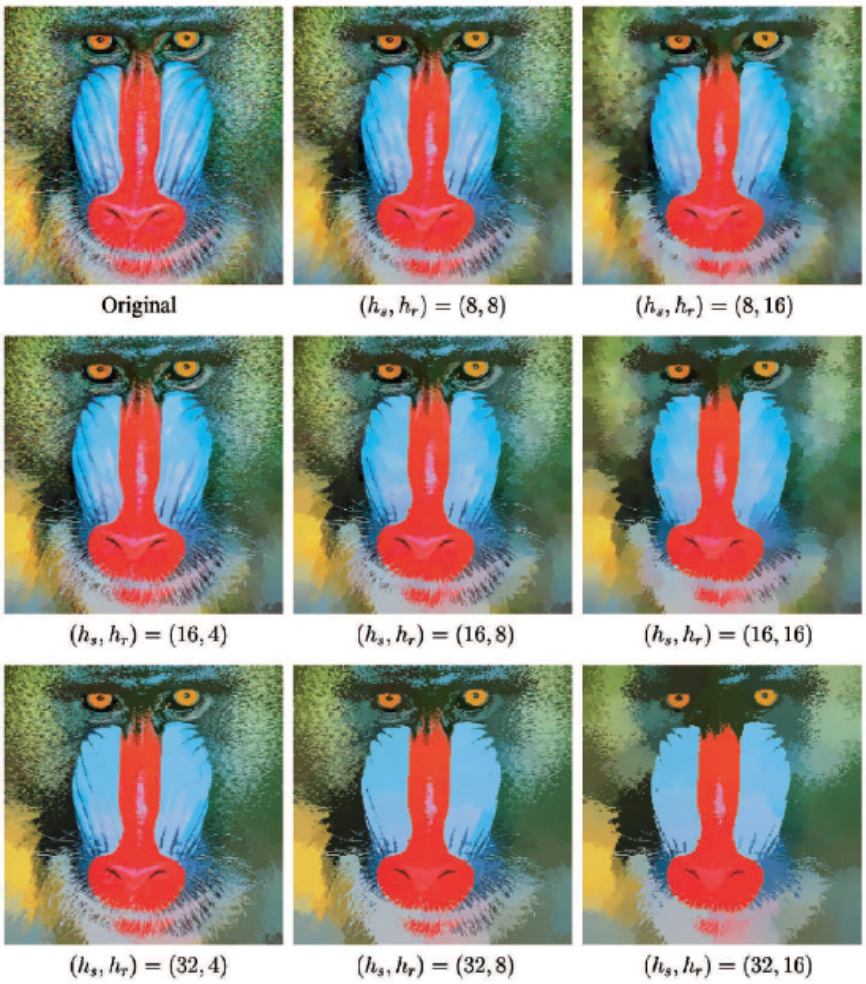


Figure 3.5: Mean shift filtered baboon image. Each color represent a segment. Image courtesy of Comaniciu and Meer [2002].

algorithms is that it finds the right number of classes based on the input image and input parameters. However, in more specialized cases, the number of classes can be a priori known. In this case the mean shift segmentation is not suited or has at least to be modified.

In Chapter 6, we specifically address the low-level segmentation task. In that chapter, mean shift is applied to segment images based on pixel values.

3.3 Semantic Image Segmentation

Semantic image segmentation is the process of partitioning an image into different regions, where each region corresponds to a class within a predefined list of labels. The appearances of

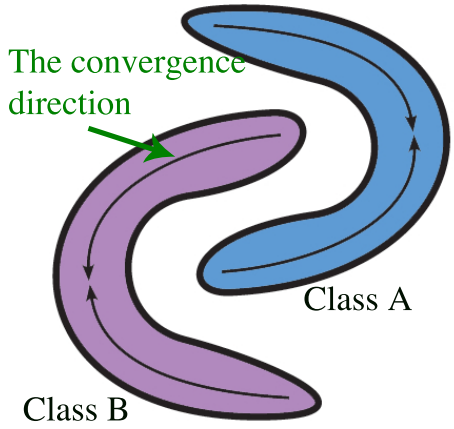


Figure 3.6: Interlaced classes in feature space.

these labels are learned during a training step on labeled images.

Recently, there have been two main trends of research in semantic segmentation:

- Development of new algorithms that can deal with both local and global characteristics of an image, e.g., based on graph theory or on feature-space analysis.
- Determination of relevant features to describe the image characteristics, e.g., color and texture representations that are both perceptually relevant and simple enough for real-time processing.

One of the lines of work is tree-based approaches, such as the semantic texton forest [Shotton et al.]. In this method, each forest is a combination of decision trees, and each tree is trained independently to predict the label for each pixel in an image. A decision tree is a directed graph, in which each node has a parent and can have many children. At each node one question is answered and a child is chosen accordingly. This process is repeated until the last generation is reached, where the node has no child and it contains the determined label of the pixel [Shotton et al.].

The segmentation can be seen as a graph labeling problem. The simplest case is the binary segmentation (foreground/background). This means that pixels need to be labeled either as positive (foreground) or negative (background). This can be trivially extended to a multi-class problem, in case we would like to segment several objects simultaneously. We assume that neighboring pixels are connected in the graph. We could consider for instance 4-connectivity or 8-connectivity. In Figure 3.7, the red circles show an example of a 4-neighborhood graph.

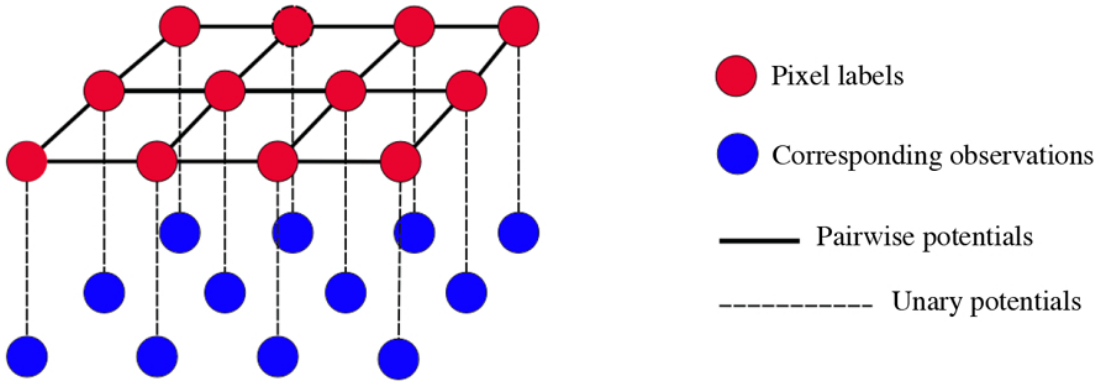


Figure 3.7: Graph over the pixels. Red nodes represent the labels of the pixels, these are estimated values. The blue nodes represent the observations. Note that observations contain information from multiple sources, and edges are associated to weights and depend on observations from neighboring pixels.

Random fields (RF): Markov Random Fields (MRF) or conditional random fields (CRF) [Kohli and Kumar, 2010, Ladicky et al., 2010] are then applied over the graph of labels. Some alternatives have been proposed to RF, for instance use of a super-pixel representation to group and regularize evidence at the pixel level. Ladicky et al. [2010] incorporate object co-occurance in CRFs, and Krähenbühl and Koltun [2011] propose fully connected CRF models defined on the complete set of pixels in an image.

In CRF-based semantic segmentation systems, we represent a pixel (or patch) i with a random variable X_i taking a value from the set of labels $\mathscr{L} = \{l_1, \ldots, l_n\}$, n being the number of classes. Let X be the set of variables representing the image and $X = \mathbf{x}$ a possible labeling of it.

The posterior distribution $P(X = \mathbf{x} \mid \mathbf{D})$, given the observation $\mathbf{D}$ over all possible labelings of a CRF is a Gibbs distribution and can be written as:

$$P(X = \mathbf{x} \mid \mathbf{D}) = \frac{1}{Z} \exp(- \underbrace{\sum_{c \in C} \psi_c(\mathbf{x}^c)}_{E(\mathbf{x})}) \tag{3.6}$$

where $\psi_c(\mathbf{x}^c)$ are potential functions over the variables $x_i^c = x_i$ for $i \in c$, and Z is a normalization factor. In this equation a clique c is a set of random variables $X_c \subseteq X$ that depend on each other, and C is the set of all cliques.

Accordingly, Gibbs energy is formed. The energy function is applied over the graph of labels. Each possible labeling corresponds to an energy value. The model is designed so that the labeling producing the minimum energy is the solution of our problem. Gibbs energy is

defined as:

$$E(\mathbf{x}) = -\log(P(X = \mathbf{x} \mid \mathbf{D})) - \log Z \tag{3.7}$$

Providing these notations, the goal of semantic segmentation is to find the most probable labeling $\mathbf{x}^*$, which is defined as the maximum a posteriori (MAP) labeling:

$$\mathbf{x}^* = \text{argmax}_{\mathbf{x}\in\mathscr{L}} P(X = \mathbf{x} \mid \mathbf{D}) = \text{argmin}_{\mathbf{x}\in\mathscr{L}} E(\mathbf{x}) \tag{3.8}$$

In the CRF model, the energy function $E(\mathbf{x})$ is composed of two terms, a unary potential E_{un} and a pairwise potential E_{pair} [3].

$$\begin{aligned} E(\mathbf{x}) &= E_{un}(\mathbf{x}) + \lambda E_{pair}(\mathbf{x}) \\ &= \sum_{i\in v} \psi_i(x_i) + \lambda \sum_{(i,j)\in\varepsilon} \psi_{i,j}(x_i, x_j) \end{aligned} \tag{3.9}$$

The unary part $\psi_i(x_i)$ of the CRF is defined as the negative log of the likelihood of a label x_i being assigned to pixel i. It can be computed from the local appearance model for each class.

More specifically, given a training set, we assume that the labeling takes the form of a ground truth segmentation. The procedure is as follows; patches are extracted and corresponding low-level descriptors, such as texture (filter banks), color statistics, and SIFT are computed across a training set. The collection of the descriptors are clustered with an unsupervised k-means/nearest neighborhood or Gaussian mixture model. The resulting clusters produce a code book of visual words. These vectors are inputs to a linear classifier. An example of such a method can be found in [Csurka and Perronnin, 2011]. One of the advantages of linear classifiers is computational efficiency. It is shown that the classification result of fast linear classifiers on higher-dimensional descriptors (i.e., FV) is as accurate as nonlinear classifiers on low-dimensional BOW [Chatfield et al., 2011]. A detailed comparison of these high-dimensional descriptors was performed in [Chatfield et al., 2011]; it is shown that the FV representation outperforms the others.

When segmenting an image, each pixel is represented by a surrounding patch, and the same representation as before is computed (for instance an FV). The representation is given to the learnt classifiers that predict scores for that pixel. These scores can be easily transformed into

[3] More complex models can contain higher order potentials

probabilities (see in [Perronnin and Dance, 2007]). Any other classification method could be used. For each pixel, its probability to belong to a class of interest is known for each of the classes. That is the information we use in the graph. More precisely, the unary term of the energy function is

$$E_{un}(\mathbf{x}) = \sum_{i \in v} \psi_i(x_i) = \sum_{i \in v} -\log(P(X_i = x_i \mid \mathbf{D})) \tag{3.10}$$

The pairwise term is responsible for the regularization in the model: neighboring pixels are encouraged to share the same label (avoiding noise in labeling). This term is also responsible for aligning the segmentation with the object borders. The pairwise terms $\psi_{i,j}(x_i, x_j)$ usually take the form of a Potts model.

$$\begin{aligned} E_{pair}(\mathbf{x}) &= \sum_{(i,j) \in \varepsilon} \psi_{i,j}(x_i, x_j) \\ &= \sum_{(i,j) \in v} (1 - \delta_{x_i, x_j}) \exp(-\beta \| p_i - p_j \|^2) \end{aligned} \tag{3.11}$$

Rother et al. [2004] proposed to set $\beta = \frac{1}{2 < \| p_i - p_j \|^2 >}$.

An exhaustive search of the best labels is in general intractable, but many exact or approximate optimization methods are applicable to our problem, for instance Gibbs sampling or graph-cut. More details on how the energy function can be minimized are given in Appendix A.

Semantic segmentation models can be enhanced using object detectors or global consistence models [Csurka and Perronnin, 2011] and a fully connected CRF model [Krähenbühl and Koltun, 2011]. These approaches are computationally more involved and also require some feature tunings. We restrict ourselves to an efficient method to show the potential advantage of introducing NIR in semantic image segmentation. For the recognition part, we compute FV descriptors and use a linear SVM classifier, and for the regularization part we use a standard CRF model. However, similar improvements are expected when the model is used in combination with more advanced approaches.

3.4 Conclusions

In this chapter we have reviewed the relevant literature on image classification and object segmentation tasks. For high-level classification methods, we have reviewed state-of-the-art

techniques and analyzed some of the algorithms. Finally, we have also discussed the clustering and low-level segmentation algorithms that are put to use in later chapters.

4 Visual Recognition

In this chapter, we evaluate the usefulness of near-infrared (NIR) images for cognition tasks. We will see that human recognition occurs in NIR images at a lower bitrate than in visible images. As different material classes have a consistent response in NIR images, color patterns on objects are most of the time transparent in such images. High-frequency information, however, tends to be preserved. Thus, an NIR image is easily interpreted by a human observer in more extreme cases of highly cluttered scenes or too low resolution. All this can be seen in Figure 4.1: half of the picture shows the visible spectrum image, and half shows the NIR image. The NIR response is completely independent of the object's color, but the object edges are clearly seen.

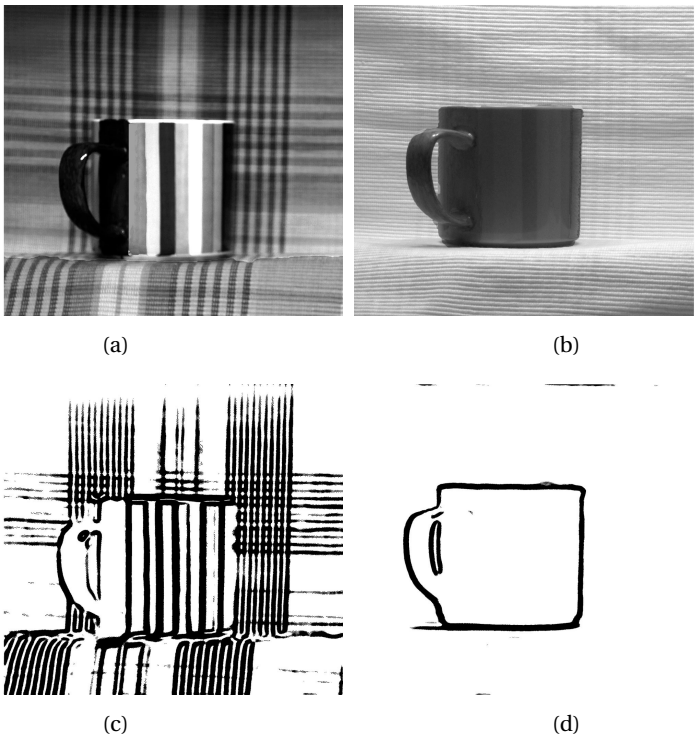
Figure 4.1: A typical photograph of a porcelain cup. (left) visible RGB image, (right) NIR image. The presence of confusing color patterns on the object makes the cognition task more difficult, especially at lower resolutions.

To evaluate the usefulness of distorted images in cognitive tasks, Rouse and Hemami [2007] introduces a *utility assessment* framework that specifies the largest distortion level that can be applied to images while still providing observers with enough information to identify the

scene. Their goal is to compare the cognition threshold for signal-based and structural-based representations of a scene, based on visible information. To generate different images in the signal-based sequence, the authors remove high frequency components of the image. The sequence of structural-based representations is formed by sequentially removing the low frequency information but keeping the edges of the image.

Utility assessment is usually studied with images that represent the scene in the visible part of the spectrum [Ullman and Power, 1997, Kosslyn, 1975]. In this chapter, our goal is to explore whether or not using representations of a scene other than visible spectrum images can facilitate the cognition task. To this end, we study how well NIR images perform in utility assessment.

Since the general shape of the objects does not change in the NIR images, edges that correspond to the object boundaries are present in these images, as can be seen in Figure 4.2. On the other hand, reflection in the NIR part of the spectrum mostly does not depend on color, so some of the patterns and fine details in the real scene disappear from the NIR images while they are perceivable in the visible images. For instance, in Figure 4.2, patterns on the cup and the background are attenuated or disappear altogether in the NIR image. As visual cognition occurs mostly based on the shape properties of the objects rather than the objects' patterns and the fine details in the scene [Ullman and Power, 1997, Sassi et al., 2010] the lack of fine details in NIR images compared to visible should not considerably affect the cognition threshold. Moreover, in some cases the presence of surface patterns in the visible image may create more difficulties in recognizing the scene. For example in Figure 4.2, the surface patterns of the cup and the background in the visible image are distracting, while in the NIR image the shape of the cup can easily be perceived.



(a) (b) (c) (d)

Figure 4.2: (a) Visible image, (b) NIR image, (c) Edge map of the smoothed visible image, (d) Edge map of the smoothed NIR image. The edge maps were produced by using difference of Gaussians on the low-pass filtered image.

4.1 Methodology

We apply the signal-based representation framework proposed in Rouse and Hemami [2007] to generate the sequence of distorted images for visible and NIR representations, and conduct a subjective test to measure the human cognition threshold for these representations.

We acquire 22 images of size 412 × 412, which represent 11 different natural scenes in the visible (luma channel only) and NIR parts of the spectrum. Figure 4.3 shows the images and a description for each image. The scenes are chosen to be similar to the images used in Rouse and Hemami [2007]. The sequence of distorted images is generated with JPEG2000 image compression [Taubman et al., 2002], which is fairly consistent with human perception. JPEG2000 mostly removes highly textured regions in the image [Sheikh et al., 2005], which are not considered useful information for cognition [Ullman and Power, 1997].

The bitrates of the compressed images were at first chosen to be logarithmically equally spaced between 0.017 and 0.765 bpp, to generate 16 images (as as been proposed in Rouse

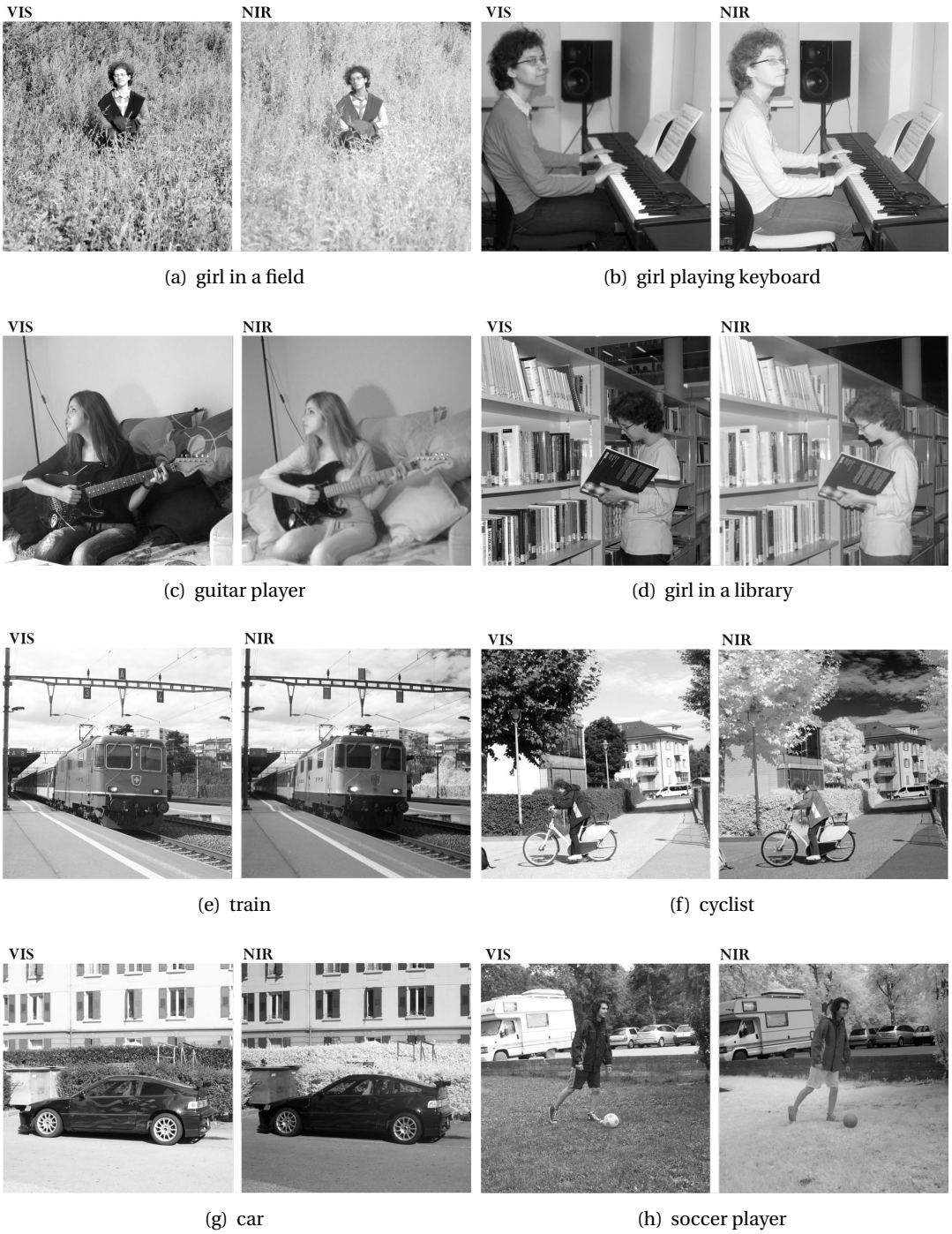
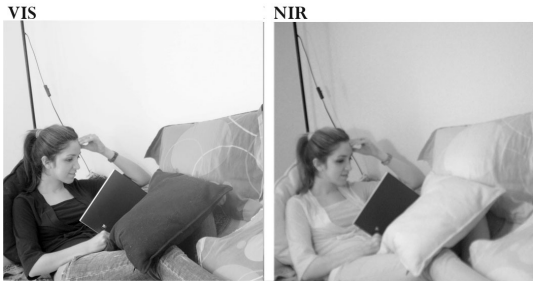


Figure 4.3: The visible and NIR representations of the scenes used in the test.

(i) boat (j) building



(k) girl reading book

Figure 4.3: The visible and NIR representations of the scenes used in the test (cont.).

and Hemami [2007]). After conducting a preliminary test, however, we find that this results in the same cognition thresholds for different observers. To increase our accuracy, we then generate 31 images with the following bitrates and use those in our main experiment:

bitrate/bpp $\in$ { 0.0170, 0.0201, 0.0210, 0.0222, 0.0229, 0.0234, 0.0239, 0.0246, 0.0278, 0.0307, 0.0327, 0.0344, 0.0393, 0.0457, 0.0550, 0.0644, 0.0774, 0.0920, 0.1078, 0.1255, 0.1511, 0.1815, 0.2041, 0.2396, 0.2864, 0.3411, 0.4078, 0.4863, 0.5481, 0.6480, 0.7600 }

Figure 4.4 shows some images in that sequence.

A subjective test is conducted to identify the observers' cognition threshold for both types of representations. Since there are two representations of each scene (NIR and visible), it is important to make sure that each subject observes only one type of representation per scene. Thus, observers who view the NIR image of a specific scene do not judge the corresponding visible image and vice versa. Each observer views a set of 31 compressed versions of each scene, which evolves from very low quality to the highest quality. At each version, they are asked to provide a general description of the scene. No time limit is imposed in the test. The bitrate of the first image from which the observer is able to describe the scene correctly is recorded as the cognition threshold for each scene and observer. The average cognition bitrate

for each scene and all the observers is computed as the cognition threshold for that scene.

30 observers (11 female, 19 male) with normal or corrected-to-normal acuity participate in the test, with ages from 24 to 39. Thus, each representation of the scene is viewed by 15 observers. A 24" display (Apple LED cinema display) with a resolution of 1920 × 1200 pixels is used to display the images, and normal office lighting is used in the test environment. The observers are seated at a distance of 40cm from the display. The 412 × 412 images are displayed on a gray background ($R = G = B = 128$).

4.2 Results and Discussion

To study the difference between the cognition threshold for NIR and visible images, we first analyze the mean cognition bitrate (M_i) for both the NIR and visible representations of all scenes in the database.

$$M_k = \frac{\sum_{i=1}^{N} m_{k,i}}{N} \qquad k \in \{\text{NIR}, \text{VIS}\} \tag{4.1}$$

where $N = 15$ is the number of valid observations for evaluating the k representation. $m_{k,i}$ is the cognition threshold for evaluation of the k representation by the i^{th} observer. Figure 4.5 shows M_k for all the images in the database. The confidence interval for each image, as shown in Figure 4.5, corresponds to the 95% confidence level. For 10 images, the NIR representation has a cognition threshold at a lower bitrate than the visible representation. We apply the *analysis of variance* ($ANOVA$) test [Snedecor and Cochran, 1989] in order to find out if the difference of the mean cognition bitrate is statistically significant. The smaller the p, the more significant the difference between the cognition bitrate of NIR and visible images. A typical value for rejecting the hypothesis that the scores for different representations come from the same population is $p < 0.05$. As noted in Table 4.1, for 8 out of the 11 images in the database, observers are able to accurately describe the scene in the NIR representation at a bitrate that is statistically significantly smaller than required for the visible images. For the "soccer player" (h) and "boat" (i) images, although the cognition bitrates of the NIR images are lower than the cognition threshold for the visible images, the difference is not significant. We notice that for the "soccer player" image, the observers were mainly focusing on the central part of the images and could not recognize the ball and the pose of the soccer player's foot, thus they could not come up with the exact description of "soccer player" at low bitrates for the NIR or visible representations. In the case of the "boat" image, the visible representation does not contain many fine details. Therefore, both the NIR and visible image have little

(a) 0.020 bpp (b) 0.026 bpp

(c) 0.039 bpp (d) 0.067 bpp

Figure 4.4: Four of 31 images of the scene “guitar player” with bitrate increasing from 0.020 to 0.067 bpp (top: NIR image, bottom: visible image).

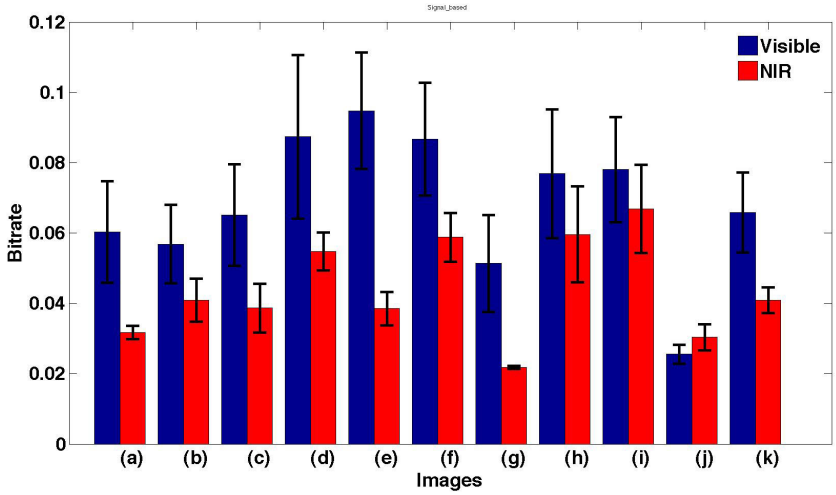


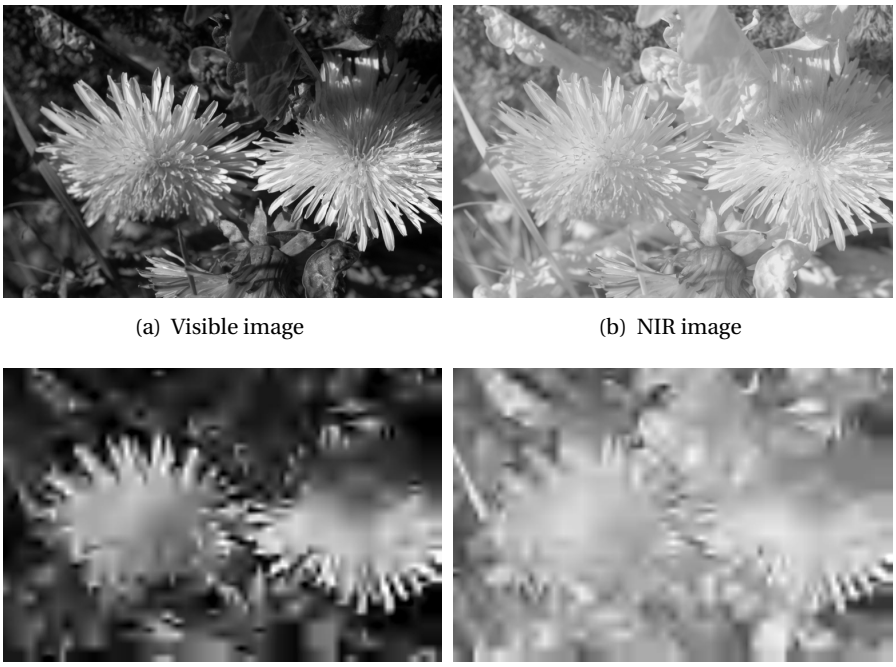
Figure 4.5: Mean cognition bitrate for both visible and NIR representation of all the images in the database.

high-frequency information and their cognition thresholds do not differ significantly. As can be seen in Figure 4.5, it is easier to recognize the "building" scene (image (j)) in the visible representation. However, the difference in cognition thresholds in NIR and visible images is not statistically significant.

Image label	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
p value	0.002	0.036	0.007	0.022	0.000	0.009	0.001	0.190	0.314	0.072	0.0011

Table 4.1: p value for all the image pairs in the database.

These results suggest that in many cases it is easier to recognize scenes from distorted NIR images than visible images. However, when different colors (or intensities) within the same material are essential for scene recognition, the NIR representation might fail. For instance, a scene containing a flower surrounded by other vegetation would be recognizable in the visible image even if the image is highly distorted (see Figure 4.6). Flower and grass have the same chemical composition and appear the same in the NIR images. Thus, choosing the NIR or the visible representation in cognition tasks depends on the specific application. While in many situations compressed NIR images are more easily recognized than their visible counterparts, the appropriateness of their material dependency needs to be evaluated for a given scenario.



(a) Visible image (b) NIR image

(c) Compressed visible image (d) Compressed NIR image

Figure 4.6: A typical photograph with vegetation. Flower and grass have the same chemical characteristics and appear the same in the NIR image.

4.3 Conclusion

We have examined the usefulness of the NIR representation in cognition tasks and compared the cognition thresholds of NIR and visible images. The cognition thresholds in NIR and visible images were evaluated using a sequence of compressed images. The results of the subjective test verified the hypothesis that for scene recognition, in many cases the NIR image is a better representation than a visible image. By extending this result to automatic scene recognition and object detection, we expected that recognizing those classes of objects will be easier for computers when NIR information is available.

5 Incorporating NIR in Image Classification

In this chapter, we tackle the scene categorization problem in the context of images where both standard visible RGB channels and near-infrared (NIR) information are available. Using efficient local patch-based Fisher Vector (FV) image representations, we show, based on thorough experimental studies, the benefit of using this new type of data. We investigate which image descriptors are relevant, and how to best combine them. In particular, our experiments show that when combining texture and color information, computed on visible and near-infrared channels, late fusion is the best performing strategy and outperforms the state-of-the-art categorization methods on RGB-only data.

Scene recognition is a long-standing problem in computer vision, being an important element in contextual vision [Heitz and Koller, 2008, Torralba, 2003, Qiang et al.]. Scene recognition capabilities are also beginning to appear in digital cameras, where "Intelligent Scene Recognition" modules can help in the selection of appropriate aperture, shutter speed and white balance.

Scene recognition is a core task of computer vision. Various methods have been proposed using different types of descriptors [Oliva and Torralba, 2001, Fei-Fei and Perona, 2005, Vogel and Schiele, 2007, Lazebnik et al., 2006, Xiao et al., 2010]. The recent study of Xiao et al. [2010] shows evidence in favour of patch-based local descriptors for color images, which has also been extended by Brown and Süsstrunk [2011] to the context of Color+NIR images.

This chapter contributes an extensive study on how NIR information can be efficiently integrated in scene categorization frameworks. Experiments are conducted on a recently released [Brown and Süsstrunk, 2011] Color+NIR semantic categorisation dataset. When relevant, some of our observations are also confirmed on two additional visible-only datasets.

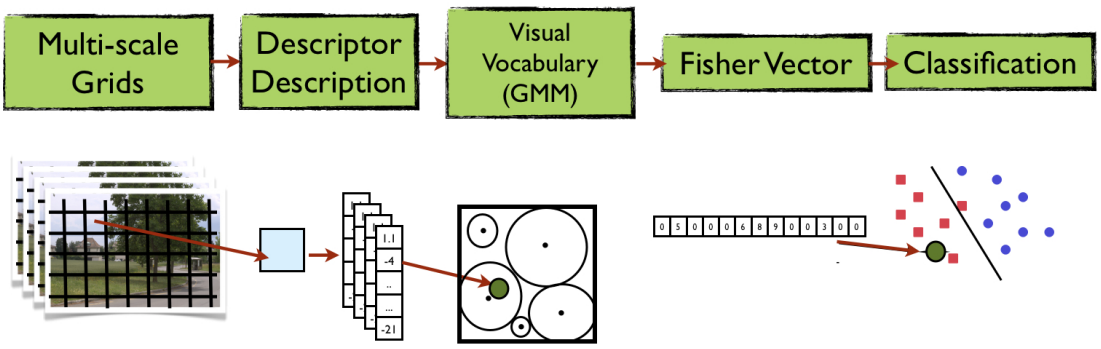


Figure 5.1: The proposed framework overview.

In our study, we apply a well-performing and generic categorisation method that works as follows: Texture or color local descriptors are encoded using Fisher Vectors [Perronnin and Dance, 2007] to produce image signatures that are then fed into discriminative classifiers. We show evidence that the categorisation method we use is highly competitive, for both Color+NIR images and standard datasets, making it a suitable framework for our study. Our contributions are three-fold: First we confirm the observation that, consistent with previous work, the combination of both color and NIR cues can be useful for image categorisation tasks on a state-of-the art pipeline. Second, we propose a thorough study on how to compute and best use texture and color descriptors when NIR information is available. Third, we investigate the complementarity between the different descriptors considered and propose efficient ways to combine them. This chapter is based on Salamati et al. [2011b].

5.1 The Proposed Approach

For our study, we use the Fisher Vector as an image representation (see Section 3.1.2 for more details). We vary the number of Gaussians in the visual codebook from 64 to 2048 and observe very little difference between classification scores, for both SIFT and color descriptors. We set 128 Gaussians for all our experiments on NIR. Similarly, for the PASCAL dataset we obtain no more improvement after 256 Gaussians. Consequently, we used this number for the two visible datasets.

During training, signatures from all training images are used to train a classifier. This classifier is then applied to each testing image signature. We use linear SVMs with a hinge loss as classifier and stochastic gradient descent (SGD) algorithm [Bottou] to optimise it in the primal formulation. The framework is fully discussed in Section 3.1, and an overview of this framework

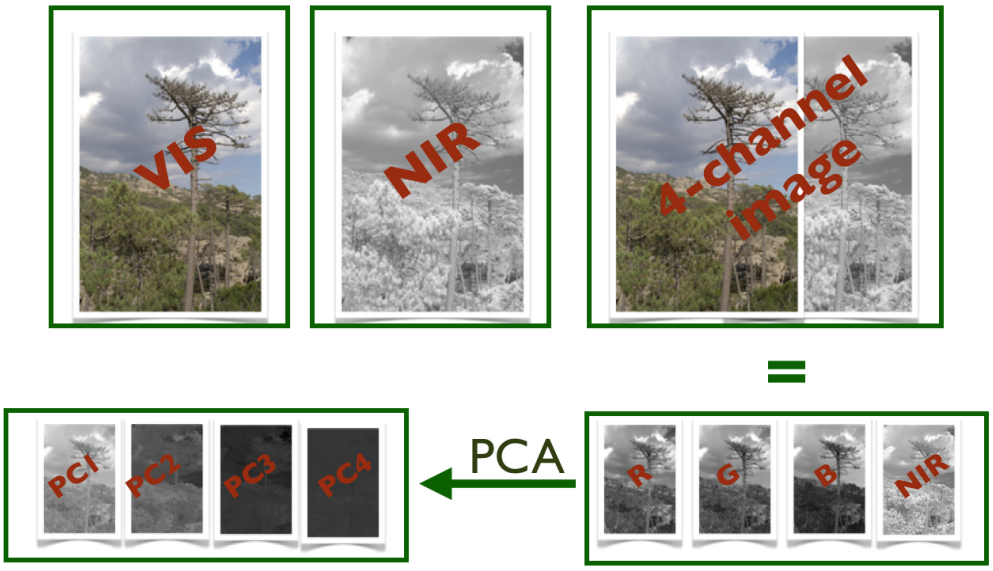


Figure 5.2: RGB-NIR color components of a scene and the corresponding channels in the PC-space. Note that there is visibly less energy in the later components.

is illustrated in Figure 5.1.

Image Channels and Feature Vectors. We use two types of feature vectors. First we consider the popular $SIFT$ [Lowe, 2004] descriptor. We use the classical 4-by-4 bins decomposition of the patch, with 8 orientation histograms computed for each bin. The concatenation of these histograms leads to a 128-dimensional descriptor (see Section 3.1 for more details).

The second type of feature vector looks directly at the intensity values in each image channel [Perronnin et al., 2010]. For a given channel, we similarly consider a 4-by-4 bin decomposition of the patch, and each bin is described by the mean and standard deviation of the intensity of pixels in that bin. This produces a 32-dimensional vector per channel, and we obtain the final feature vector, called COL, by concatenating the vectors of the relevant channels.

For visible-only datasets, images are described by 3 color channels: r, g and b. For the NIR dataset, the original images contain 4 different channels, r, g, and b for the visible part, and a near-infrared (NIR) channel denoted by n. We additionally consider the luminance channel denoted by l, which can be computed on the visible part (r,g,b).

As in Brown and Süsstrunk [2011], we consider an alternative way of dividing images into channels. The initial 4D (r,g,b,n) vector is decorrelated and a PCA projection is computed.

	r	g	b	n	$p1$	$p2$	$p3$	$p4$	l
$SIFT$	$SIFT_r$	$SIFT_g$	$SIFT_b$	$SIFT_n$	$SIFT_{p1}$	$SIFT_{p2}$	$SIFT_{p3}$	$SIFT_{p4}$	$SIFT_l$
COL	COL_r	COL_g	COL_b	COL_n	COL_{p1}	COL_{p2}	COL_{p3}	COL_{p4}	-

Table 5.1: Summary of basic features considered for combination.

Afterwards, each pixel can be described in this new four-dimensional space, as $p1$, $p2$, $p3$, $p4$. This new color space can replace the conventional channels. This projection takes place such that the first component, i.e., $p1$, explains the maximum amount of variance of the data. In other words, the first component captures the largest amount of information in the decomposition (see Figure 5.2 for an example). Brown and Süsstrunk [2011] showed that it is composed of a positive and almost equal contribution of all 4 channels.

The SIFT and COL descriptors can be applied and combined in different ways on these channels, hence we obtain a large set of possible features for building the image signatures. Table 5.1 summarises the basic features we consider for further combinations. We use the following notations. $SIFT_{i,j,k}$ denotes the concatenation of the $SIFT_i$, $SIFT_j$ and $SIFT_k$ features, computed on the channels i, j and k respectively. Similarly, $COL_{i,j,k}$ concatenates the COL_i, COL_j and COL_k descriptors into a 96-dimensional descriptor. For 2 and 4 channels (such as $SIFT_{l,n}$ and $COL_{r,g,b,n}$) similar rules are applied.

5.2 Datasets

Our experimental study is conducted primarily on a recently released dataset, the **EPFL scene-classification dataset (EPFL)** [Brown and Süsstrunk, 2011], containing images that are composed of visible and infrared channels.

Some of our observations can be transposed to standard color images. In these cases, we show additional evidence on two other datasets, composed only of traditional visible (RGB) images. The first one is another scene dataset (MIT-Scene 8), and the other one is a standard object classification dataset (PASCAL VOC 2007).

Near Infrared Dataset. The EPFL scene-classification dataset consists of 477 images, divided into 9 scene categories (Country, Field, Forest, Mountain, Old building, Street, Urban, Water, Indoor). Although the number of images in this dataset is limited, it contains challenging classes. Some classes are fine-grained and contain several common elements: e.g., Old building (Figure 5.3(d)) and Urban (Figure 5.3(e)) are overlapping, or Country (Figure 5.3(c)) could be confused with Water. Also, this is the only available benchmark dataset for categorisation

that contains both RGB and NIR channels for each image.

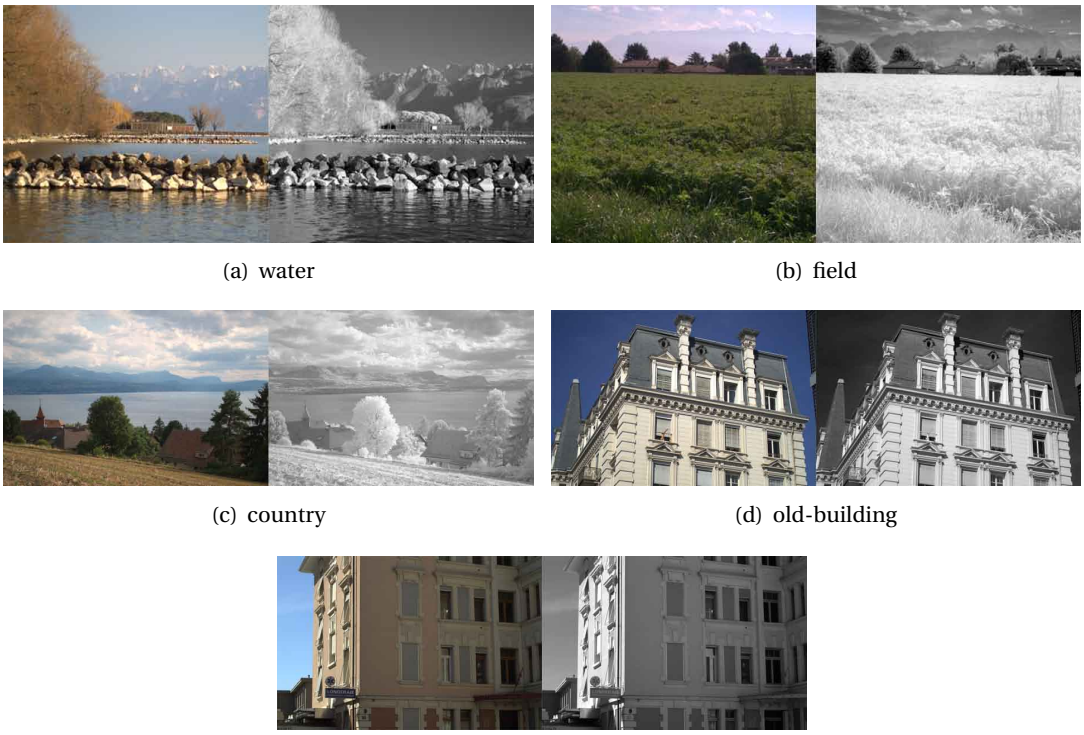
As described in Brown and Süsstrunk [2011], the dataset was acquired as follows: To capture the NIR channel, the NIR blocking filter in the digital camera has been removed. As such modified cameras do not allow for a joint acquisition, two separate exposures have been captured, a visible only (with NIR blocking filter) and a NIR one (without NIR filter and with a filter which suppresses visible light). The possible movements between the two shots are corrected using an alignment algorithm. At the end, images composed of 4 channels are produced. More details on the transformation between NIR sensor responses and a single NIR channel can be found in Fredembach and Süsstrunk [2008]. Examples of the RGB and NIR channels on some images are displayed as pairs in Figure 5.3. On the left, a conventional RGB image of a natural scene is shown, the right side shows the NIR representation. Note that, as expected, NIR images generally have similar appearances, although there are some differences, e.g., bright vegetation and dark sky and water in the NIR channel.

For evaluation, we follow the same protocol as Brown and Süsstrunk [2011]: We randomly select for testing 11 images per class (99 total) and train the classifiers using the remaining images. As they did, we repeat this process 10 times, and we report the mean and standard deviation of the classification accuracy.

Visible Benchmark Datasets. We also use the following two categorisation datasets.

First, we use the **MIT scene-8 classification dataset (MIT)** [Oliva and Torralba] as it is composed of similar scene categories as the EPFL dataset. It contains 2688 images of 8 outdoor scene categories: Coast, Mountain, Forest, Open country, Street, Inside city, Tall buildings and Highways. Following the work of Oliva and Torralba, we randomly select 100 images per class for training, and the remaining images are used for testing. We repeat this process 5 times, and report the mean and standard deviation of the average accuracy.

We also test our observations on the popular **PASCAL VOC Challenge 2007 dataset (PASCAL)** [Everingham et al., b] to show that similar conclusions can be reached for object categorisation. It contains 9963 images and 20 object classes. In our experiments, we use the provided split in a training set and a testing set, and we compute mean average precision (MAP) over the classes. This allows us to fairly compare our results with the state of the art.



(a) water (b) field

(c) country (d) old-building

(e) urban

Figure 5.3: Some images of the EPFL dataset are displayed as pairs. On the left: the conventional RGB image of the scene, on the right: its NIR counterpart.

5.3 Experimental Study

In this section we describe the experiments we conducted. First, we show the positive influence of applying the principal component analysis (PCA) to both local descriptors. Then, we analyse the behaviour of the SIFT and color descriptors independently. Finally, in order to determine the best suited strategy, different combinations of these descriptors are discussed.

5.3.1 The Influence of PCA on Local Features

In the state-of-the-art framework, a PCA projection is usually applied to reduce the dimensionality of the local descriptors. Here, we study the influence of this step and compare three configurations: i) no PCA is applied and the original descriptors are directly used to build the codebook, ii) the PCA is used in order to reduce the dimensionality to D=64 (as in Perronnin et al. [2010]), and iii) the PCA is applied and descriptors are projected into the PC space, but the full dimensionality is kept. For the latter case, referred to as Full PCA, the projected feature vectors have the same dimensionality as the original descriptors. For “visible-only”

	Descriptor	$SIFT_l$	$COL_{r,g,b}$
PASCAL	Orig. desc.	51.6	37.6
	PCA with 64D	59.4	48.2
	Full PCA	59.4	48.2
MIT	Orig. desc.	90.1 ± (0.6)	75.8 ±(1.1)
	PCA with 64D	92.1 ±(0.2)	86.9 ±(0.3)
	Full PCA	92.2 ±(0.2)	86.9 ± (0.4)
EPFL	Orig. desc.	79.1 ± (4.7)	71.6 ± (2.6)
	PCA with 64D	83.4 ± (2.6)	80.2 ± (4.1)
	Full PCA	83.4 ± (2.8)	81.7 ± (2.8)

	Descriptor	$SIFT_{p1,p2,p3,p4}$	$COL_{r,g,b,n}$
EPFL	Orig. desc.	83.8 ± (3.4)	71.1 ± (3.9)
	PCA with 64D	85.1 ± (3.4)	81.7 ± (1.8)
	Full PCA	85.9 ± (4.0)	82.2 ± (1.6)

Table 5.2: Mean average precision (MAP) for PASCAL, and class accuracy (mean ± std) for MIT and EPFL, with different PCA configurations.

descriptors (i.e., $SIFT_l$ and $COL_{r,g,b}$ that use only RGB channels), the results are reported in the first three sub-tables of Table 5.2 for all three datasets. To study the PCA effect when the NIR channel is available, we also evaluate $SIFT_{p1,p2,p3,p4}$, used in Brown and Süsstrunk [2011], and $COL_{r,g,b,n}$, as a direct extension of $COL_{r,g,b}$. The results are shown in the last sub-table of Table 5.2.

First, we confirm that this PCA step has a crucial role and is needed in the framework. If we compare Full PCA to the original descriptors, for visible-only feature vectors, the difference is of a few percent for $SIFT_l$ (7.8% for PASCAL, 2.2% for MIT, and 4.3% for EPFL). For $COL_{r,g,b}$, differences are larger, about 10%. Similar results are achieved when the NIR channel is available (few percent for $SIFT_{p1,p2,p3,p4}$ and 11.1% for $COL_{r,g,b,n}$).

We now look at PCA as a dimensionality reduction method. Table 5.2 reports results for a reduction to 64 dimensions. We also compare the final accuracy for a wide range of PCA projection dimensions, for both RGB+NIR descriptors (Figure 5.4). Results show that increasing the PCA dimensions only very rarely decreases the accuracy, and we consistently observed the highest mean accuracy for full-dimension PCA. Therefore, we choose to use the full resolution descriptors on the PCA space for the rest of our tests. Nevertheless, the stability observed over the set of dimensions indicates that PCA can also be used to reduce the feature vector dimension and speed up the computations for larger-scale experiments.

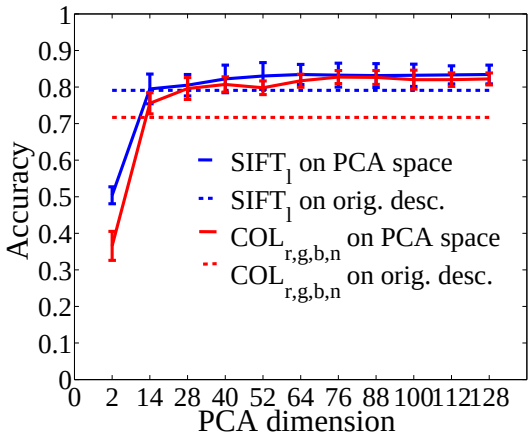


Figure 5.4: EPFL dataset: varying PCA dimensions for $SIFT_i$ and $COL_{r,g,b,n}$

5.3.2 Local Descriptors Study

We begin our study by looking at individual descriptors. As mentioned earlier, we consider two complementary feature vectors. The SIFT descriptor encodes gradient orientation, and consequently captures both the image contour, and the texture contained in a patch (Section 5.3.2). As color information is also a characteristic of scenes, and as an additional channel (NIR) is available, we then look at the influence of the simple color descriptor COL. We will show evidence of their complementarity in Section 5.3.3.

SIFT Features

In this section, we investigate different ways of computing SIFT descriptors on RGB+NIR images. The standard SIFT feature is $SIFT_l$, which considers the visible-only luminance channel. $SIFT_n$ describes the NIR sensor response and is an obvious candidate for evaluation. We also consider $SIFT_{p1}$ as, by construction, the channel $p1$ incorporates a large amount of information from r,g,b and n. Finally, we combine the first two descriptors, $SIFT_l$ and $SIFT_n$ by concatenating them, as they contain non-overlapping information. Classification accuracies are shown in Table 5.3.

First, we observe that all these descriptors perform quite similarly. As expected, $SIFT_{p1}$ has a rather similar performance with $SIFT_l$, as it is achromatic and contains a positive and almost equal contribution of all 4 (as opposed to 3 for $SIFT_l$) original channels. More surprisingly, $SIFT_n$ performs similarly, whereas SIFT computed on any single visible color channel performs worse (e.g., $SIFT_g$ gets 82.0% ± (2.8)). Nevertheless, the loss is not too important, which shows that some texture information can be kept in any of these channels.

	Descriptor	$SIFT_l$	$SIFT_{l,n}$	$SIFT_n$	$SIFT_{p1}$	$SIFT_{p1,p2,p3,p4}$	$SIFT_{r,g,b,n}$
EPFL	Accuracy	83.4 ± (2.6)	84.3 ± (3.3)	83.7 ± (2.4)	83.0 ± (2.3)	85.9 ± (4.0)	82.7 ± (3.1)

Table 5.3: Accuracy (mean ± std) on EPFL for SIFT features extracted on different channels.

This is further confirmed on the 2 other datasets (on MIT for instance, $SIFT_l$ is 92.2 ± (0.2), while $SIFT_r$ gives 91.9 ± (0.3)).

$SIFT_{l,n}$, the concatenation of $SIFT_l$ and $SIFT_n$, gives an accuracy of 84.3%, a small improvement over their individual results and thus confirming that they indeed contain complementary information.

Instead of using a dedicated color descriptor, as we will do in the next section, the color information can be included in SIFT by concatenating SIFT descriptors per channel over the full 4D space. Results for $SIFT_{r,g,b,n}$ and $SIFT_{p1,p2,p3,p4}$ are also reported in Table 5.3. $SIFT_{r,g,b,n}$ performs equally, and not better than, luminance SIFT, although it uses additional NIR information. $SIFT_{p1,p2,p3,p4}$, the multi-spectral SIFT descriptor proposed by Brown and Süsstrunk [2011], stands out with 85.9% accuracy. By using a more adapted color space, it makes better use of both texture and color information.

We could have compared these with other versions of the color SIFT descriptor, as proposed by van de Sande et al. [2010], but Brown and Süsstrunk [2011] already showed that $SIFT_{p1,p2,p3,p4}$ performs better than them.

Color Features

Another feature relevant to classification is color, which looks at the relation between r, g, b, and n channel values. In this part, we compare 4 different color descriptors. $COL_{r,g,b}$ is the standard one and considers only visible information. Its direct extension to the NIR domain is $COL_{r,g,b,n}$. We also consider RGB+NIR information in the alternative color PCA space, $COL_{p1,p2,p3,p4}$. Finally, as the last channel of this descriptor contains little information and a significant amount of noise, we also look at the reduced $COL_{p1,p2,p3}$ descriptor. Table 5.4 compares their classification accuracy.

The first observation is that good performance is obtained, even though these color descriptors encode only simple statistics. For instance, $COL_{r,g,b,n}$ is only 1.5% below $SIFT_n$ in the EPFL set. $COL_{r,g,b}$ on visible datasets, although not competetive with SIFT, still achieves fair performance, considering the simplicity of this descriptor (48.2% on PASCAL and 86.9% on MIT).

	Descriptor	$COL_{r,g,b}$	$COL_{r,g,b,n}$	$COL_{p1,p2,p3,p4}$	$COL_{p1,p2,p3}$
EPFL	Accuracy	81.7 ± (2.8)	82.2 ± (1.6)	83.0 ± (2.2)	83.2 ± (2.4)

Table 5.4: Accuracy (mean ± std) on EPFL, color features on different channel combinations.

Table 5.4 also confirms that incorporating NIR information increases the classification accuracy. The visible baseline $COL_{r,g,b}$ is slightly below $COL_{r,g,b,n}$, and is clearly outperformed by $COL_{p1,p2,p3,p4}$. This shows that the PCA projection in the color space, which de-correlates the channels r, g, b, and n, further improves the results of the COL descriptor.

Finally, we can also observe that removing $p4$, which contains the least amount of information, does not change the classification accuracy, while slightly reducing the computation cost (smaller feature dimension).

5.3.3 Fusion of SIFT and Color Information

Now that we have studied SIFT and color descriptors independently, we are interested in strategies that combine texture and color information. We already considered the $SIFT_{r,g,b,n}$ and the more successful $SIFT_{p1,p2,p3,p4}$, which encode both by using only the SIFT descriptor on multiple channels. In the following, we show that we can go beyond these accuracies by combining the previously studied SIFT and COL features.

We consider two types of combination. *Early fusion* is done at the descriptor level, i.e., we concatenate both extracted features for each patch, and then apply a full PCA projection on the concatenated descriptors. *Late fusion* combines the features at the latest stage by averaging the classifier outputs obtained for both descriptors. In this study, we use equal weights for late fusion, because optimisation of the weights on a validation set would be difficult, given our already small training set.

For SIFT, we keep $SIFT_{l,n}$ and $SIFT_n$ descriptors as they obtain best performances in the first part of our SIFT study. For the color feature we test all features considered in 5.3.2, to see how well they compliment those two SIFT features. Table 5.5 shows the performance of early fusion (EF) and late fusion (LF) strategies on all 8 possible SIFT/COL pairs. We can draw several observations from this table.

First, it is clear that late fusion always outperforms early fusion. The difference is often significant (3% on average). Similar results are also observed for the two visible-only datasets. For MIT and PASCAL datasets, early fusion of $SIFT_l$ and $COL_{r,g,b}$ achieves 90.7% (± 0.4) and 54.9%, whereas late fusion obtains 92.4% (± 0.6) and 61%, respectively. This could be explained

Classifier	Descriptor1	Descriptor2	Fusion type	Accuracy
FV + SVM, with NIR	$COL_{r,g,b}$	$SIFT_{l,n}$	EF	84.3 ± (1.7)
			LF	**87.4 ± (2.7)**
		$SIFT_n$	EF	84.1 ± (2.6)
			LF	86.2 ± (2.0)
	$COL_{r,g,b,n}$	$SIFT_{l,n}$	EF	84.4 ± (2.8)
			LF	85.5 ± (2.1)
		$SIFT_n$	EF	84.1 ± (2.9)
			LF	86.5 ± (2.4)
	$COL_{p1,p2,p3}$	$SIFT_{l,n}$	EF	83.3 ± (3.4)
			LF	86.7 ± (2.7)
		$SIFT_n$	EF	82.9 ± (2.7)
			LF	**87.5 ± (2.4)**
	$COL_{p1,p2,p3,p4}$	$SIFT_{l,n}$	EF	85.8 ± (2.6)
			LF	86.5 ± (2.8)
		$SIFT_n$	EF	83.2 ± (3.0)
			LF	**87.9 ± (2.2)**
FV + SVM, no NIR	$COL_{r,g,b}$	$SIFT_l$	LF	84.5 ± (2.3)
Brown and Süsstrunk [2011]	–	–	–	72.0 ± (2.9)

Table 5.5: Accuracy (mean ± std) for different fusions on the EPFL dataset

by the very different nature of the combined descriptors. Building specific classifiers for each descriptor, and doing a combination at the decision level, is therefore superior.

Second, we can see that all the late fusion combinations, for which at least one descriptor contains NIR cues, outperform the "visible-only" baseline, i.e., the results obtained by the late fusion of $SIFT_l$ with $COL_{r,g,b}$. This again underlines the usefulness of the NIR information for this categorisation task.

Finally, we observe the benefit obtained by the proposed strategy: the fusion of independently trained SIFT and color descriptor-based signatures gives up to 87.9% accuracy, and outperforms the multi-spectral SIFT ($SIFT_{p1,p2,p3,p4}$) at only 85.9%. The best results are obtained with $SIFT_n + COL_{p1,p2,p3,p4}$, but $SIFT_n + COL_{p1,p2,p3}$ and $SIFT_{l,n} + COL_{r,g,b}$ yield very similar performances. For visible only datasets, the same observation holds: $SIFT_{r,g,b}$ obtains 86.9% and 54.1%, and the late fusion of $SIFT_l$ and $COL_{r,g,b}$ achieves 92.4% and 61.0% (MIT and PASCAL, respectively). This shows that for both standard and RGB+NIR datasets, color information can improve SIFT based classifiers the most when used as a specific descriptor in its own pipeline and combined with late fusion.

Comparison with Previous Work. To improve our results, as the geometry is usually expected to be consistent across scene categories, we combine the FV representation with the spatial pyramid (SP) technique [Lazebnik et al., 2006], as suggested in Perronnin et al. [2010]. For the visible datasets, we apply SP to $SIFT_l$ and $COL_{r,g,b}$ independently, and we combine

them with late fusion. The results[1] are indeed further improved on MIT, from 92.4% to 93.6% and on PASCAL, from 61.0% to 63.7%.

However, when we apply SP on the EPFL dataset, on $COL_{p1,p2,p3}$ and $SIFT_n$, their late fusion gave 87.5%, which is similar to its no-SP counterpart (87.5%). The absence of improvement can be justified by the limited size of our training set.

All reported results significantly outperform the 72% (± 2.9) on the EPFL dataset reported in Brown and Süsstrunk [2011]. This difference is partly justified by the FV framework we used. Using the same local descriptor as Brown and Süsstrunk [2011], i.e., $SIFT_{p1,p2,p3,p4}$ reduced to 128D, already leads to a much higher accuracy (85.1 (± 3.4)). Nevertheless, our experiments confirm that the idea of de-correlating the four color channels allows us to improve the categorisation accuracy. Based on this color space, our best strategy obtains a final accuracy of 87.9%.

For the MIT and PASCAL datasets, we obtain results similar or better than the state of the art. On the MIT dataset, Oliva and Torralba report a classification accuracy of 83.7% with GIST features. Our best result, 92.4%, was obtained with $SIFT_l + COL_{r,g,b}$ (LF) and SP. On the PASCAL dataset we obtain 63.7%, which is comparable to the 64.0% of Zhou et al. [2010].

5.4 Conclusion

In this chapter, we have presented a thorough study of the scene categorisation problem, in the case where NIR information, which can be captured by a normal digital camera, is available in addition to visible light. This study is based on the Fisher Vector representation, a generic and powerful categorisation framework. As a conclusion specific to that method, we have shown the usefulness of applying a PCA projection to local descriptors.

Through the study of two generic local descriptors, we have obtained NIR specific conclusions. First, we have shown that NIR is a useful piece of information that, combined with visible cues, can improve recognition. In particular, the SIFT descriptor in the NIR domain performs similar to the standard luminance-based SIFT. Both are outperformed by the concatenation of SIFT descriptors computed on each channel of the alternative PCA color space ($p1, p2, p3, p4$).

Second, we have also investigated the best way to include the 4D color information in our categorisation method. We propose using a color descriptor that encodes local statistics about

[1] As SP increases the image signature dimensionality, we reduce the feature vectors to 64D for these last experiments.

color information and the NIR channel. This simple descriptor performs almost as well as the SIFT alternative, in particular in the alternative PCA color space instead of the conventional r, g, b, and n space. The late fusion of FV signatures computed on the best color descriptor ($COL_{p1,p2,p3,p4}$) and on the NIR SIFT descriptor ($SIFT_n$) is shown to be the best categorisation strategy, and outperforms multi-channel SIFT descriptors.

This observation generalises to visible datasets, showing that color information can be better used in a specific descriptor. A classifier trained on color-descriptor based signatures, combined by late fusion with a complementary SIFT based classifier, outperforms the multi-channel SIFT descriptor ($SIFT_{r,g,b}$).

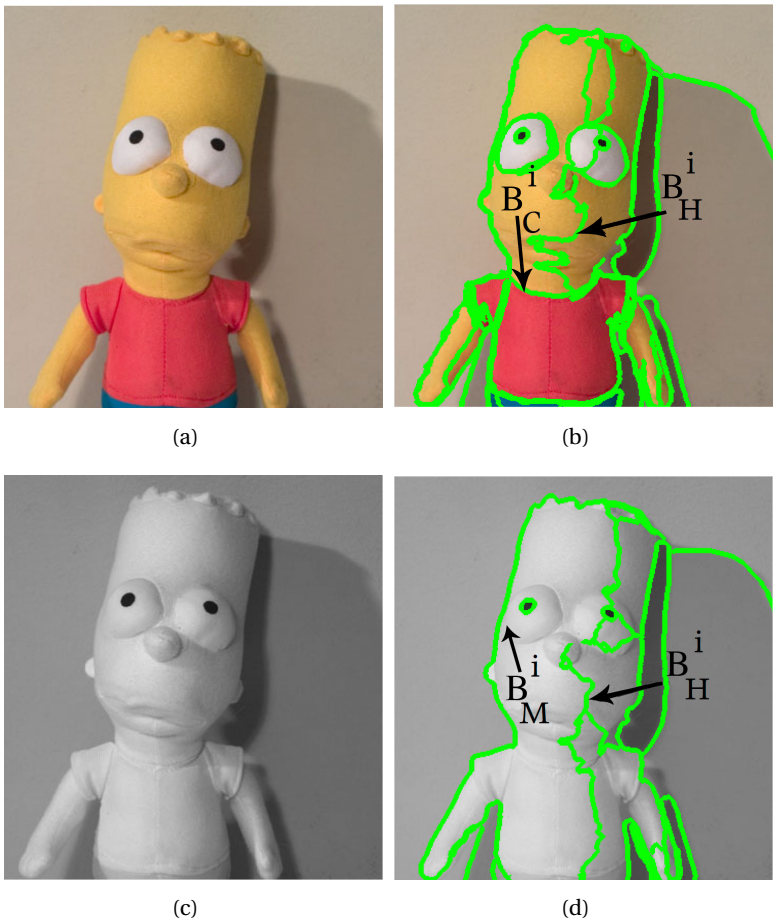
6 Material-Based Boundary Detection

In this chapter we determine where material changes occur in an image, based on low-level features, i.e., we segment the image so that the segment boundaries correspond only to object boundaries.

NIR radiation generally penetrates deeper than visible light into an object's surface and can reveal the underlying material characteristics [Burns and Ciurczak, 2001, Pohl and Van G., 1998]. As such, changes in intensity in the NIR image are due to material and illumination changes, but not to color variations within the same material (see Figure 6.1 for illustration).

One source of mis-segmentation are shadows that occur due to the shape of the object and/or the geometric arrangement of the object and the light source. Many algorithms have been proposed to correct the color values in an image so that the edges corresponding to shadows are not confounded with object boundaries [Brill, 1990, G.D. Finlayson and Lu, 2006]. Some shadow removal frameworks try to recover an image based on the ratios of color bands, in which the absolute intensity variation over an object is reduced and the result is invariant to shadows [Funt and Finlayson, 1995].

Inspired by the 4-sensor camera calibration model by Finlayson and Drew [2001], we combine both visible RGB and near-infrared (NIR) images to obtain an *intrinsic image* that is independent of illumination. Different pixel values represent reflectance variations (thus color and material changes), but they are shadow-independent. The union of the NIR and intrinsic image segmentations results in segment borders that are only caused by material changes, but not by color and shadow variations within the object. This chapter is based on Salamati and Süsstrunk [2010].



(a) (b)

(c) (d)

Figure 6.1: The mean shift segmentation result of visible and NIR images. The first row shows the color image and its segmentation result, the second row is the NIR image and its segmentation. Note the oversegmentation resulting from changes in illumination ${B_H}^i$ or colors ${B_C}^i$ within the object.

6.1 Our Proposed Approach

Figure 6.1 shows the segmentation result of a visible image and its NIR counterpart. As illustrated in Figure 6.1 (b), the segment borders in the visible image B_v are due to changes in color, as well as due to areas with different shadows and shading:

$$B_v = B_C \cup B_H \tag{6.1}$$

where B_C is the set of all the borders due to color changes and B_H is the set of borders between areas in which the illumination is different.

In NIR images, changes in material, shading, and cast shadows are responsible for the borders B_n (see Figure 6.1 (d) for illustration). Thus,

$$B_n = B_M \cup B_H \tag{6.2}$$

where B_M is the set of all borders due to material changes.

To achieve an accurate material-based segmentation, we need to eliminate the borders generated due to different illumination conditions and to varying colors within the same material. Our proposed approach involves creating an image independent of the lighting conditions, i.e., an *intrinsic image*. Its segment borders B_i are therefore due to changes in material or color only.

$$B_i = B_M \cup B_C \tag{6.3}$$

We then derive our material-based segmentation by applying the " $\cap$ " operator to the NIR and intrinsic image segmentation results:

$$B_n \cap B_i = (B_M \cup B_H) \cap (B_M \cup B_C) = B_M \cup (B_H \cap B_C) \simeq B_M \tag{6.4}$$

The term $B_H \cap B_C$ is usually not significant, because an illumination-based border and a color border are unlikely to randomly coincide for more than a few pixels. However, in cases where both B_H and B_C coincide due to the object's geometry (e.g., the cube in Figure 6.2 with the sides colored differently has both color and shading borders at its edges), our approach will incorrectly oversegment the picture.

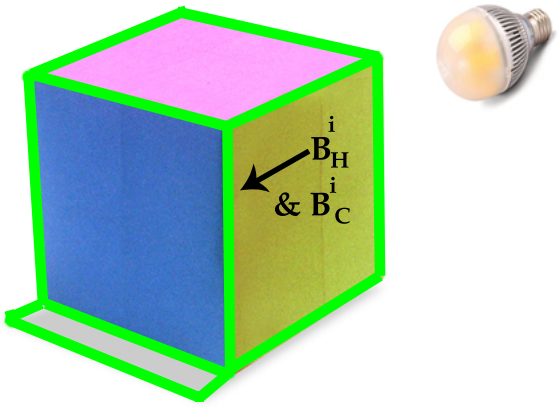


Figure 6.2: An example of an object, in which an illumination-based border and a color border coincide.

6.2 The Physical Properties of Visible and NIR Signals

Visible and NIR image intensities depend on the interaction between the surface properties of the object, illuminants, and the camera. The color signal $C(\lambda)$ describes the influence of the scene, i.e., object and illuminants:

$$C(\lambda) = S(\lambda) \times E(\lambda) \tag{6.5}$$

where $S(\lambda)$ is the reflectance of the surface and $E(\lambda)$ is the illuminant spectral power distribution. Figure 6.3 depicts three different relations that can hold between the signals $C(\lambda)$ of different regions in an image and their interpretation.

The sensor response I_k of a sensor k with sensitivity R_k can be simplified as

$$I_k = \int_{\lambda=400}^{1100} C(\lambda) \times R_k(\lambda)\, d\lambda = \int_{\lambda=400}^{1100} S(\lambda) \times E(\lambda) \times R_k(\lambda)\, d\lambda \tag{6.6}$$

We use $k \in \{R, G, B, NIR\}$ for the 4 channels available to us.

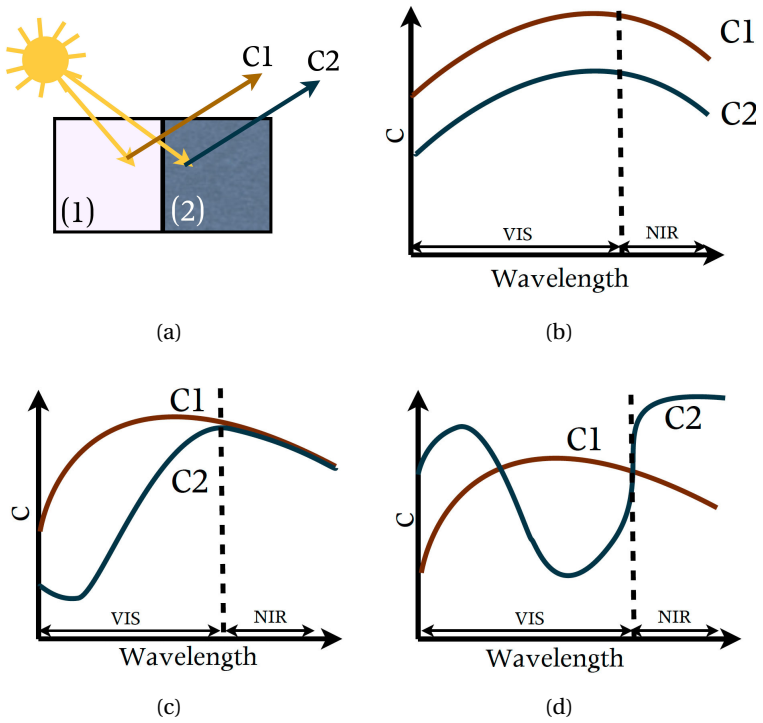


Figure 6.3: Three different relations that can hold between the color signals $C(\lambda)$ of two regions in an image. (b) Region (2) is under a shadow, (c) (1) and (2) are of the same material but colored differently, and (d) a color and material change occurs.

Depending on the location of the object with respect to the camera and the light source, a shadow can be cast. This shadow results in a reduction of measured intensity. As a first approximation, we describe the image intensity in the lit and shadow part of the object I_k^{lit}, I_k^{shade} as follows (see Figure 6.3 (b) for illustration): If

$$I_k^{lit} = \int_{\lambda=400}^{1100} S(\lambda) \times E(\lambda) \times R_k(\lambda)\, d\lambda \tag{6.7}$$

then the shadow part of that object is described as

$$I_k^{shade} = \int_{\lambda=400}^{1100} S(\lambda) \times aE(\lambda) \times R_k(\lambda)\, d\lambda \tag{6.8}$$

where a represents a fraction of the light intensity ($0 \leq a \leq 1$).
We assume thereby that the ambient light (illuminating the shadowed parts of the object) shares its spectral characteristics with the main light source. Although this assumption does not hold in general, it has successfully been applied in other color correction models [Levine

and Bhattacharyya, 2005]. We then consider the following relations and find that the ratio of the I_{RGB} to the I_{NIR} response across a material with a certain color stays unaffected by shadows:

$[{I^{shade}}_R, {I^{shade}}_G, {I^{shade}}_B] = a[{I^{lit}}_R, {I^{lit}}_G, {I^{lit}}_B]$ and

$${I^{shade}}_{NIR} = a{I^{lit}}_{NIR}$$

$$\Longrightarrow \frac{[{I^{lit}}_R, {I^{lit}}_G, {I^{lit}}_B]}{{I^{lit}}_{NIR}} = \frac{[{I^{shade}}_R, {I^{shade}}_G, {I^{shade}}_B]}{{I^{shade}}_{NIR}} \tag{6.9}$$

The second interesting case is when both regions belong to the same material, but are colored differently (see Figure 6.3 (c) for illustration). NIR imaging is 'transparent' to a number of colorants and dyes; it can see through the first layer and reveal the material surface underneath [Salamati et al., 2009]. Thus, the NIR images reveal more information about the material an object is made of.

Each class of material has an affinity towards a certain class of colorants (chemistry and functional bond specific), due to the chemistry and the process of coloring different materials. Hence, even if the object colors are not transparent to the NIR spectrum, the NIR response is likely to be the same (${I^{(1)}}_{NIR} = {I^{(2)}}_{NIR}$) [Burns and Ciurczak, 2001], because the different colorants used to dye the material are probably chemically similar. Consequently,

$$\frac{[{I^{(1)}}_R, {I^{(1)}}_G, {I^{(1)}}_B]}{{I^{(1)}}_{NIR}} \neq \frac{[{I^{(2)}}_R, {I^{(2)}}_G, {I^{(2)}}_B]}{{I^{(2)}}_{NIR}}. \tag{6.10}$$

This assumption does not always hold, particularly for very dark dyes that tend to also absorb in NIR.

Figure 6.3 (d) shows a change in both color and material. In this case, the ratio of I_{RGB} to I_{NIR} is not constant for the two patches (see Equation 6.10).

6.3 Forming the Intrinsic Image

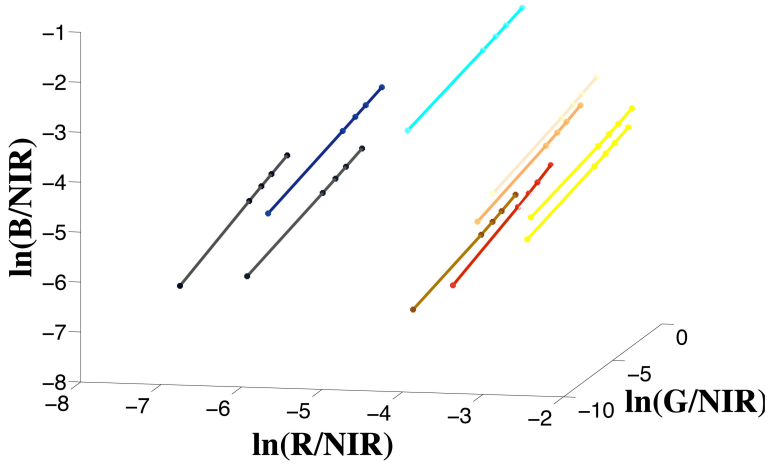
Up to now, we have argued that R, G, and B to NIR ratio images are potentially able to present changes that correspond to either different materials or different colors within that material. Inspired by the physics of the NIR and color signals of a surface, we modify the algorithm by Finlayson and Drew [2001]. Their color-constancy-at-a-pixel algorithm tries to find the coordinates in which the ratio image is invariant to both intensity and color of the illuminant. It is based on the assumption that the illuminants can be modeled as having black-body spectra. It also assumes sensors receptive to a single wavelength λ_k only, i.e., the sensor response can be modeled by the Dirac delta function. With these assumptions, the logarithmic response of sensor k for an illuminant $E(\lambda, T)$ can be calculated:

$$
\begin{aligned}
&E(\lambda_k, T) = K_1 {\lambda_k}^{-5} e^{-\frac{K_2}{T\lambda_k}} \\
&I_k = \int_{\lambda=400}^{1100} S(\lambda) \times E(\lambda, T) \times \delta_{\lambda_k}(\lambda)\, d\lambda = S(\lambda_k) \times E(\lambda_k, T) \\
&\Longrightarrow log(I_k) = -\frac{1}{T}(\underbrace{\frac{K_2}{\lambda_k}}_{E_k}) + \underbrace{log\left(K_1 \lambda^{-5} S(\lambda_k)\right)}_{S_k}
\end{aligned}
\tag{6.11}
$$

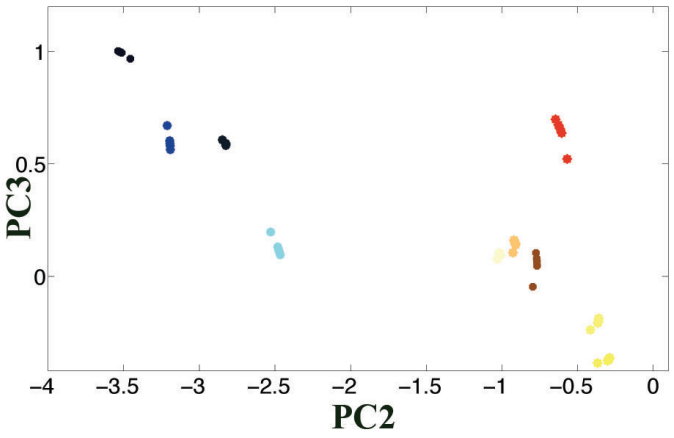
Here, $S(\lambda_k)$ is the reflectance of the surface being imaged at wavelength λ_k, T is the color temperature of the light, and K_1 and K_2 are constants. The first term in this equation, $-\frac{1}{T}E_k$, depends on the illuminant's color temperature, and the last part, S_k, depends on the surface reflectance. Given 4 sensors $k \in \{k_1, k_2, k_3, k_4\}$, subtracting the response of one logarithmic sensor from those of the other 3 sensors gives us the equation of a line in 3-dimensional space in which the reflectance dependent part appears as the intercept and the illuminant dependent part is the slope of the line.

$$
\begin{aligned}
log(\frac{I_{k_1}}{I_{k_4}}) &= log(I_{k_1}) - log(I_{k_4}) = S_{k_1} - S_{k_4} - \frac{1}{T}(E_{k_1} - E_{k_4}) \\
log(\frac{I_{k_2}}{I_{k_4}}) &= log(I_{k_2}) - log(I_{k_4}) = S_{k_2} - S_{k_4} - \frac{1}{T}(E_{k_2} - E_{k_4}) \\
log(\frac{I_{k_3}}{I_{k_4}}) &= log(I_{k_3}) - log(I_{k_4}) = S_{k_3} - S_{k_4} - \frac{1}{T}(E_{k_3} - E_{k_4})
\end{aligned}
\tag{6.12}
$$

Thus, adjusting the color temperature of the light source T changes the log-ratio of the sensor responses along a single direction. This 3-dimensional space can be projected onto



(a)



(b)

Figure 6.4: (a) The log ratio of 10 samples under different light sources/shadows. The intensity ratio of all the samples under different lights lies along a single direction, (b) The chromaticity space given by the projection onto the second and third principle eigenvectors.

a 2-dimensional space where illuminant induced variation is minimized, i.e., the new 2-dimensional representation of any image will be independent of the illuminant's color temperature. As we assumed that the illuminant's only characteristics are intensity and color temperature and we eliminated both, we now have an illumination-independent representation.

To find the actual numbers for this conversion on our camera, we measure the reflectances of

50 objects in the visible and NIR spectrum. Together with the known wavelengths for the 4 camera sensors, they allow us to calculate theoretical sensor responses I_k, $k \in \{R, G, B, NIR\}$ under different Planckian light sources with temperatures of 3000, 5000, 6000, 6500 Kelvin, as well as under equi-energy lighting.

The log-ratios, R_{IR}, B_{IR} and G_{IR}, are then computed as follows:

$$R_{IR} = ln(\frac{I_R}{I_{NIR}}), \;\; G_{IR} = ln(\frac{I_G}{I_{NIR}}), \;\; B_{IR} = ln(\frac{I_B}{I_{NIR}}) \tag{6.13}$$

Figure 6.4 (a) shows the log ratio of 10 samples under 6 different illuminants. All the intensity ratios of the samples under different lights roughly lie along a single direction. For all the samples under different light sources, the covariance matrix

$$cov = \begin{bmatrix} cov(R,R) & cov(R,G) & cov(R,B) \\ cov(G,R) & cov(G,G) & cov(G,B) \\ cov(B,R) & cov(B,G) & cov(B,B) \end{bmatrix} \tag{6.14}$$

can then be computed. The best coordinate system is found by the eigenvectors of the covariance matrix. For our database, the eigenvectors are:

$$\mathbf{C} = \begin{bmatrix} 0.378 & 0.89 & 0.23 \\ 0.54 & 0.00 & -0.84 \\ 0.75 & -0.44 & 0.49 \end{bmatrix}$$

The samples' log-ratios are projected onto the two eigenvectors with smaller eigenvalues using the following equation:

$$\begin{bmatrix} PC2 \\ PC3 \end{bmatrix} = \begin{bmatrix} 0.89 & 0.00 & -0.44 \\ 0.23 & -0.84 & 0.49 \end{bmatrix} \times \begin{bmatrix} R_{IR} \\ G_{IR} \\ B_{IR} \end{bmatrix} \tag{6.15}$$

Figure 6.4 (b) shows the samples in the database in the new space. In this space, each sample under a specific light source appears as a dot, and the same sample under other light sources projects to approximately the same position. Applying Equation 6.13 and 6.15 at each pixel position produces the desired *intrinsic image*, see Figure 6.5 for some examples. Figure 6.5 (c) and (e) are taken under an unknown illuminant. Their results, however, are fairly invariant to the light source's intensity. This can be explained by the object reflectances in the NIR part of

the spectrum, as discussed in Section 6.2.

The primary drawback associated with this approach is its inability to differentiate dark plastic objects situated close to brighter objects (as illustrated in Figure 6.5 (h) where a black object is placed in front of the grey background or the white parts of the doll). Carbon black is used as a pigment in rubber and dark plastic products (polymers in general). This pigment reflects almost no light in both the visible and the NIR part of the spectrum and therefore appears dark in both images. Thus, the shadow relation can hold between the black object and brighter grey or white surroundings, and both will be mapped onto the same value.

6.4 Our Segmentation Procedure

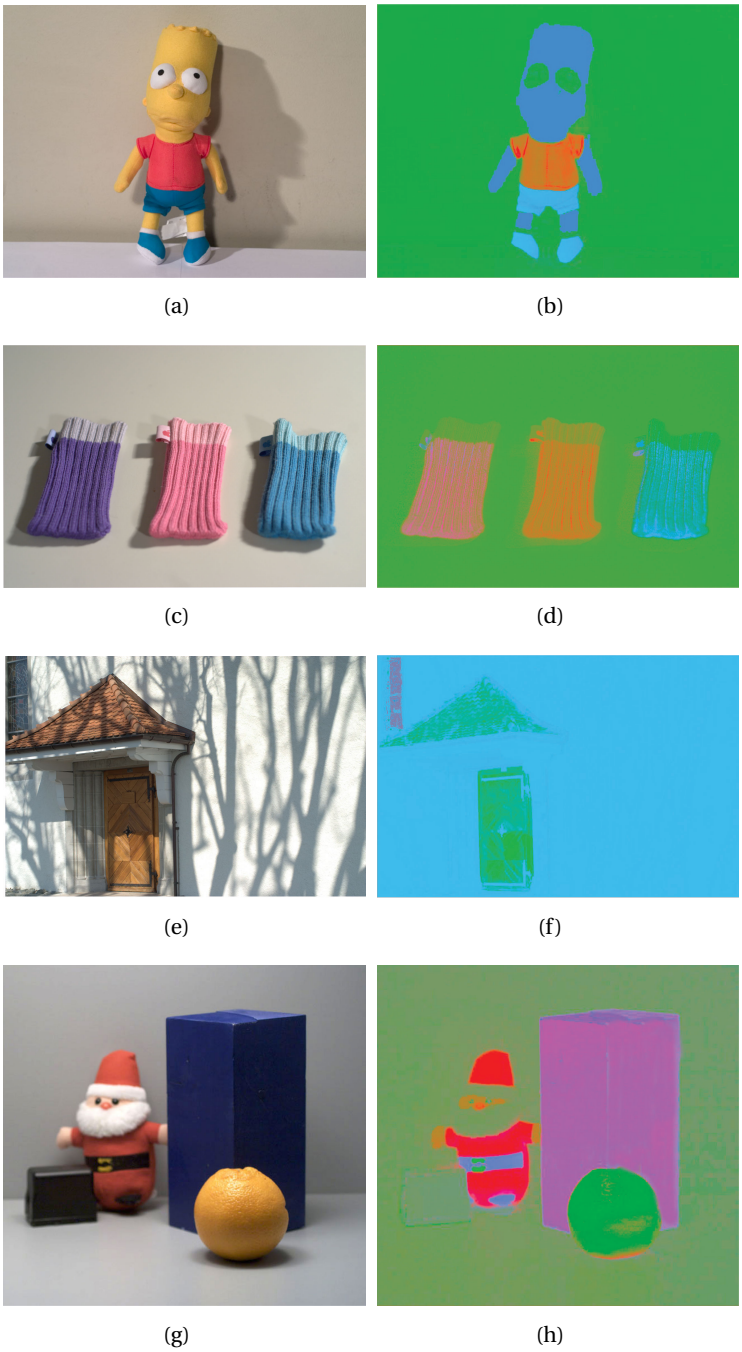
The idea is to segment the illuminant-independent images as well as the NIR images. As it has already been formulated in Equation 6.2 and 6.3, segment borders B_n in the NIR images are formed due to changes in material or in the illuminant, and segment borders B_i in the intrinsic images are formed due to changes in material or color. Thus, logically, the physical object boundaries are those present in both images.

To segment the images, the mean shift algorithm is applied to both intrinsic and NIR images. It is an image clustering method based on color and spatial features [Comaniciu and Meer, 2002]. The main idea behind the algorithm is to compute for every single pixel a series of mean values in feature space. The mean is shifted towards more densely populated regions in the feature space. Each segment contains all data points in the attraction basin of a convergence point. This approach does not require a priori knowledge of the number of segments (see Section 3.2.1 for more details).

The feature space for segmenting NIR images is the pixel intensity, and for illuminant-invariant images, $PC2$ and $PC3$ coordinates form the feature vector. A key feature to make our implementation work is the detection of *all* the boundaries corresponding to material changes. To ensure this even for the (single-channel) NIR image segmentation, the resolution parameters for the mean shift algorithm are chosen 10% larger than those for the intrinsic images.

After segmentation, the borders of all segments form a binary edge map. As the resolution parameters of the segmentation algorithm are different for the intrinsic and NIR images, the corresponding segment borders of these two images might not overlap. Thus, the binary edge maps are dilated with a circle of size 3 as the structuring element. The final segmentation is the result of applying the " $\cap$ " operator on the dilated edge maps of the segmented NIR and

illumination-invariant images. Figure 6.6 provides a flowchart detailing the segmentation framework.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 6.5: (First Column) visible images and (second column) the illuminant-independent representation. To visualize images in the new space, we present $PC2$ and $PC3$ as a and b values in the $CIELAB$ color space. Lightness value is chosen to be 60 for all the intrinsic images.

6.5 Results

All the images were photographed in the visible and the NIR range of the spectrum. The camera we used is a modified Canon EOS 300D [Fredembach and Süsstrunk, 2008].

We compare the mean shift visible/NIR segmentation with the mean shift on visible images only. The results are shown in Figure 6.7. This comparison provides useful insight into how accurately our segmentation procedure is able to find the physical object's boundaries. We notice that regions with a small gradient of illumination or color are successfully segmented as a single region. For instance, the orange in Figure 6.7 (e) and (i) (visible-only) is divided into different segments because of the changes in illumination, whereas in Figure 6.7 (f) and (j) (visible+NIR) the actual physical boundary of the orange is detected.

As the green object on the wall in Figure 6.7 (i) and (j) demonstrates, the precision in object boundary retrieval is higher using the proposed framework. The visible-only segmentation results are more sensitive to the resolution parameters in the mean shift algorithm, due to variations of illumination. In visible-only segmentation there is always a trade-off between accurate borders and additional segments detected within an object. With our approach, however, we can simply increase the resolution parameter to obtain the exact boundaries in both NIR and intrinsic image segmentations. By applying the "$\cap$" operator, all the undesired segments are removed.

The main drawback of our approach is the loss of the dark plastic objects in segmentation when they are set against an achromatic background. (see Figure 6.7 (f)). Another drawback of this method is the presence of some segment borders in the result that do not correspond to any changes of material (see Figure 6.6: the segmentation result of the doll around the eye and the collar). This mis-segmentation occurs when there are many variations in the channel intensities due to the illumination. Then there are be so many borders in the NIR image, corresponding to the changes of illumination, that they randomly intersect with the borders due to changes of color in the intrinsic image and form new segments. This could be mitigated by postprocessing the merged edge map or using a more sophisticated merging operator. Over-segmentation due to object geometry and corresponding coloring (see Section 6.1) seems to be unavoidable with this approach.

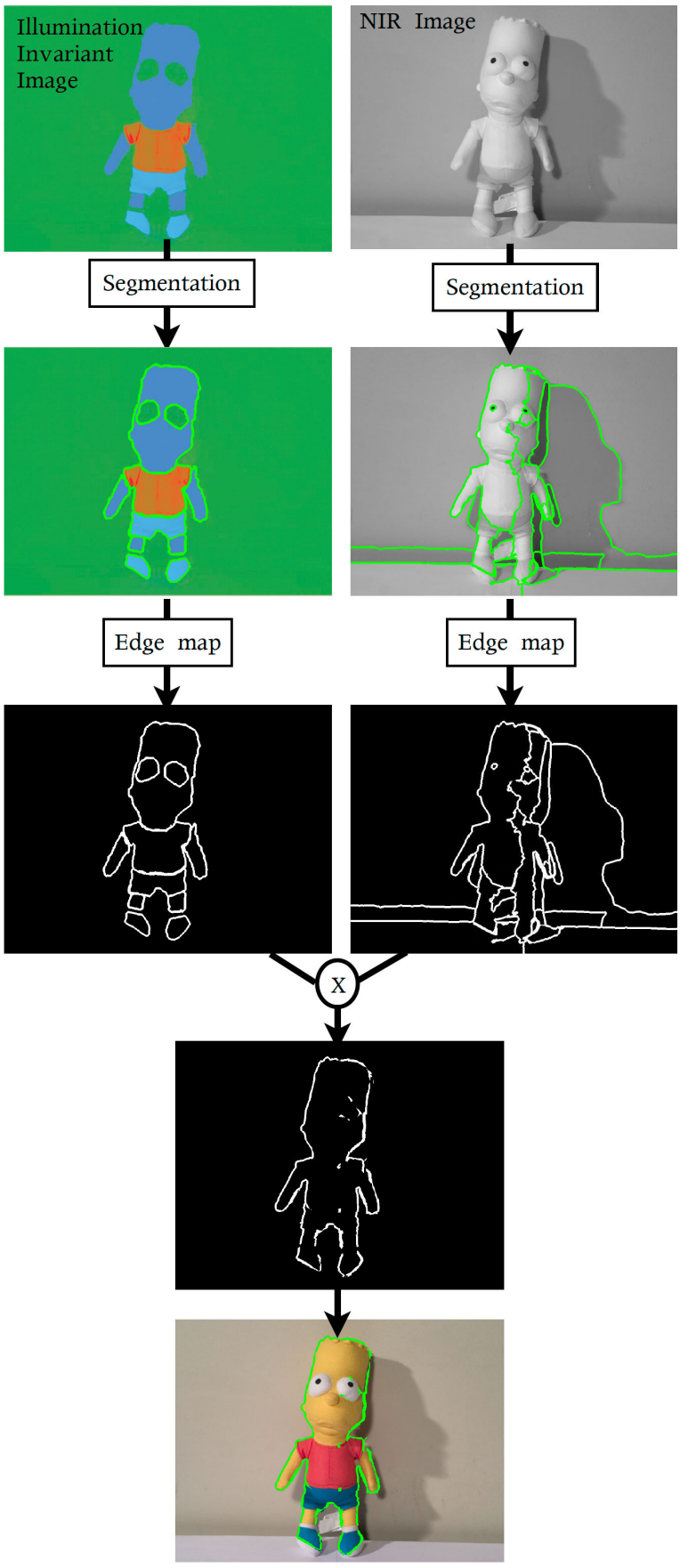
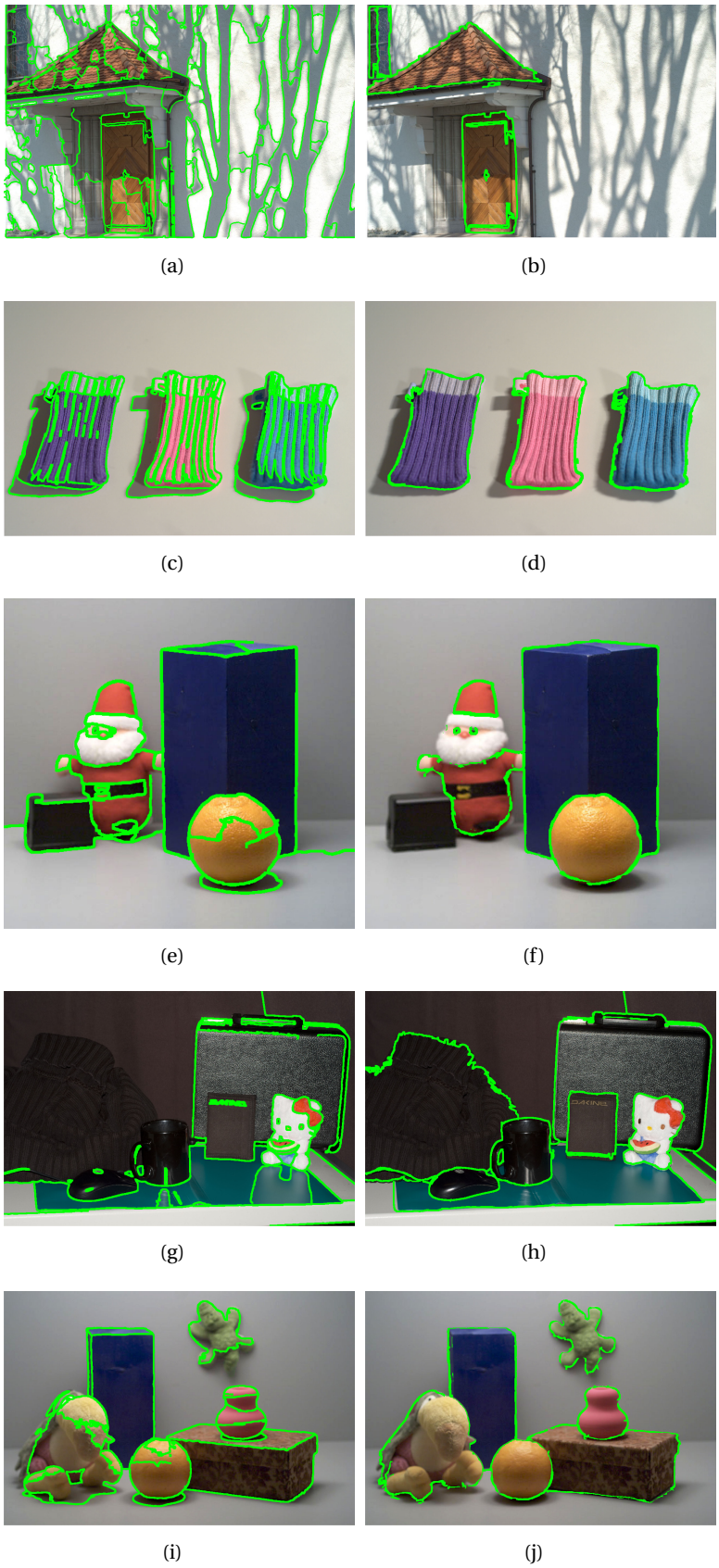


Figure 6.6: The flowchart detailing the segmentation framework.



(a) (b)

(c) (d)

(e) (f)

(g) (h)

(i) (j)

Figure 6.7: (First Column) visible-only image segmentation result, (second column) segmentation result using joint information.

6.6 Conclusion

We have presented a method that accurately detects physical object boundaries in images by using visible RGB and NIR information. In order to discard the borders corresponding to color changes within an object, we propose using NIR image segmentation only. By combining the NIR information as the fourth channel, along with RGB values, to form an illumination-independent image, we can achieve a shadow-free representation of the scene. The union of the two segmentation results produce segments that are only material dependent. By applying the proposed framework on real images, we show that segmentation using NIR information, as well as visible images, yields more accurate results in detecting physical object boundaries, especially when the object consists of one material.

In this chapter we have shown that the material-dependeny characteristics of NIR images improve the border accuracy of low-level segmentation and the detected borders correspond to the actual boundaries of the physical object. The results suggest that in order to more precisely extract the object pixel-level location, the special characteristics of NIR can be incorporated in a more sophisticated high-level segmentation framework. As the next step, to obtain a better semantic segmentation result, we propose leveraging the information available in the NIR domain. Essentially we propose combining both the visible and the NIR information in a joint graph based model. The 4-channel image is used to efficiently recognize and roughly locate the object of interest in the image. The NIR information is expected to improve the accuracy of the recognition results as well as boundary accuracy.

7 Semantic Image Segmentation

Our task is to recognize and locate, at the pixel level, different categories of objects (e.g., Tree, Building, etc.) in an image. For example, in Figure 7.1, the segmentation of a cup (image mask) is shown in the right image.

Figure 7.1: (left) input image, (right) segmentation of the cup.

Segmentation algorithms usually fail in the presence of a cluttered background or distractor objects. Even when the appearance of objects can be successfully recognized, the contour of the objects can be confused with strong edges coming from the background clutter or any distracting regions. Inhomogeneities, such as shadows, specularities, changes of color within the object and variations in pigment density, will introduce gradients in the image, which can confound segmentation algorithms; thus resulting in multiple distinct segments being assigned to one single object.

In this chapter, we consider how to avoid mis-segmentation due to variations in surface color and shadows, in both indoor and outdoor scenes. To this end, we incorporate the near-infrared (NIR) information into low-level and high-level segmentation frameworks.

We are interested in the problem of semantic segmentation, i.e., assigning each pixel in an

image to one of several semantic classes. This is a supervised learning problem in contrast to "classic" unsupervised segmentation that groups pixels into homogeneous regions based on low-level features, such as color or texture. We propose a novel graph-based energy minimization approach that combines visible and NIR information in order to produce a segmentation. In this chapter, we describe our graph-based model that captures both the visible and the NIR channels and uses the unique benefits of each channel in an appropriate manner to segment a given image. This chapter is based on Salamati et al. [2012].

7.1 Our Proposed CRF Framework

We propose to use the CRF model that is described in Section 3.3. In our CRF model, the energy function $E(\mathbf{x})$ is composed of two terms, a unary potential E_{un} and a pairwise potential E_{pair} [1]. The unary term is responsible for the recognition part of the model and the pairwise term encourages neighboring pixels to share the same label. We assign a weight λ to E_{pair} that models the trade-off between recognition and spatial regularization.

$$\begin{aligned} E(\mathbf{x}) &= E_{un}(\mathbf{x}) + \lambda E_{pair}(\mathbf{x}) \\ &= \sum_{i \in v} \psi_i(x_i) + \lambda \sum_{(i,j) \in \varepsilon} \psi_{i,j}(x_i, x_j) \end{aligned} \tag{7.1}$$

where v corresponds to the set of all image pixels and ε the set of all edges connecting the pixels $i, j \in v$ in 4×4 or 8×8 neighborhood.

In this model, both the unary and the pairwise potential are built using information extracted from the RGB images. In the following, we will show how to extend both the unary and the pairwise potential by integrating the NIR channel (or denoted for simplicity N channel) in the above energy term.

7.1.1 The Unary Term

The unary part $\psi_i(x_i)$ of the CRF is defined as the negative log of the likelihood of a label x_i being assigned to pixel i. It can be computed from the local appearance model for each class. (See Section 3.3 for more details.)

$$E_{un}(\mathbf{x}) = \sum_{i \in v} \psi_i(x_i) = \sum_{i \in v} -\log(P(X_i = x_i \mid \mathbf{D}))$$

[1] As discussed, more complex models can contain higher order potentials

The local appearance is defined at every pixel location, on a regular grid, or at interest points. To build our local appearance model, we follow Csurka and Perronnin [2011] and use patch level Fisher Vectors (FV) to train the classifiers. We then propagate the patch based predictions to obtain class probability maps.

This consists of extracting overlapping image patches on a multi-scale grid and describing them with low-level descriptors, such as SIFT. The dimension of these features can optionally be reduced by using principal component analysis (PCA) before building a Gaussian-mixture-model (GMM) based visual vocabulary. This allows us to transform, for each patch, the low-level representation into an FV (see for more details Section 3.3 and Perronnin et al. [2010], Csurka and Perronnin [2011]).

We use FV representations that encode higher order statistics than the visual word counts in the bag of visual words (BOW) representation [Csurka et al., 2004]. We choose them not only because they outperform the BOW (as shown in Csurka and Perronnin [2011]), but also because they are highly competitive for object classification even with linear classifiers [Chatfield et al., 2011]. However, the BOW or other low-level to high-level representations could also have been used.

Then, for each class, we train a patch-level linear classifier by using strongly labeled training images (segmented images), and we transform the classification scores of each patch in the test image into probabilities. At the pixel level, the class posterior probabilities are obtained as a weighted average of the patch posteriors, where the weights are given by the distance of the pixel to the center of the patch, as in Csurka and Perronnin [2011].

Usually, several types of features are used to model the appearance of a class and to infer the likelihood of a given object at a given location. The most popular descriptors for the visible-only datasets are a variation of color and texture features. Here, we consider the popular SIFT [Lowe, 2004] features to describe the local texture, and local color statistics [Clinchant et al., 2007] to describe the color. The latter, referred to here as COL, encodes the mean and standard deviation values of the intensity values in each image channel for each bin of a 4x4 grid covering the patch (same bins as in the case of $SIFT$). In our visible-baseline approach, these color statistics are computed in the R,G and B channels, hence we will denote their concatenation by COL_{rgb}.

SIFT encodes local texture with a set of histograms of oriented gradients computed on each a 4x4 grid covering the patch. In general, it is computed on the patch extracted from the luma channel of the visible RGB image that can be approximated by $L = 0,299R + 0,587G + 0,114B$.

Dataset	R-G	G-B	R-B	R-N	G-N	B-N
outdoor	99.07	98.62	97.32	93.07	93.21	88.87
Indoor	97.34	96.80	91.76	90.56	90.89	87.80

Table 7.1: Correlation ($Corr_{K,L}$) between different channels in both outdoor and indoor scenes.

It will therefore be denoted by $SIFT_l$.

We can also extract these low-level features in the N channel (computing local moments or oriented gradients histograms). We will denote them by COL_n and $SIFT_n$ respectively. They can be further concatenated with features extracted from the RGB image, leading to, for example, COL_{rgbn} or $SIFT_{rgbn}$.

Due to the high correlation of RGB and N channels, Brown and Süsstrunk [2011] show that incorporating NIR information in a de-correlated space improves the performance of image classification. Table 7.1 shows the correlation $Corr_{K,L}$ between different channels ($K, L \in$ {R,G,B,N}) in our outdoor and indoor datasets. It is computed as

$$Corr_{K,L} = \frac{\sum_i p_i^K p_i^L}{\sqrt{(\sum_i (p_i^K)^2 \times \sum_i (p_i^L)^2)}} \tag{7.2}$$

where p_i^K and p_i^L are pixel values at location i in channels K and L, respectively.

Therefore, we also consider combining VIS with NIR information in the alternative-color PCA space denoted by $COL_{p1p2p3p4}$ and $SIFT_{p1p2p3p4}$.

In Section 6, we compare and discuss the performance of using each of these descriptors in the unary term of our energy function.

7.1.2 The Pairwise Term

The pairwise terms $\psi_{i,j}(x_i, x_j)$ of our CRF take the form of a Potts model:

$$\begin{aligned} E_{pair}(\mathbf{x}) &= \sum_{(i,j)\in\varepsilon} \psi_{i,j}(x_i, x_j) \\ &= \sum_{(i,j)\in v} (1 - \delta_{x_i,x_j}) \exp(-\beta \| p_i - p_j \|^2) \end{aligned} \tag{7.3}$$

where $\delta_{x_i,x_j} = 1$ if $x_i = x_j$, and $\delta_{x_i,x_j} = 0$ otherwise. We set $\beta = \frac{1}{2<\|p_i - p_j\|^2>}$, as in the work of Rother et al. [2004]. This potential penalizes disagreeing labels in neighboring pixels, and the penalty is lower where the image intensity changes. In this way, borders between predicted regions are encouraged to follow image edges.

In general, the pixel values p_i in the Potts model correspond to the RGB colors in the test image. In this case we denote the pairwise term by VIS, as it corresponds to the visible image.

However, when NIR information is available, we can use the N channel in the pairwise potential. In this case p_i corresponds to the intensity in the N channel and the potential will be denoted by NIR. Finally, when use the intensity values from all the 4 channels (p_i is 4-dimensional) the pairwise potential is denoted by $VIS + NIR$.

7.1.3 Model Inference

Given our CRF model, we want to find the most probable labeling ($\mathbf{x}^*$), i.e., the labeling that maximizes a posteriori labeling. The MAP labeling for many practical multi-label computer vision problems is NP-hard and approximation algorithms have to be used.

For our model, the inference, using α-expansion (see Appendix A for more details), is carried out by the multi-label graph optimization library of Boykov et al. [2001], Boykov and Kolmogorov [2004], Kolmogorov and Zabin [2004]. The idea of this algorithm is to reduce the NP problem to a sequence of binary optimization problems. Given a labeling $\mathbf{x}$, each pixel i makes a binary decision to keep its current label or switch to the new label α.

7.2 Datasets and Experimental Setup

In order to evaluate and better understand the gain of incorporating NIR in different parts of our CRF model, we build two different datasets and test the model on them separately. Examples of these 4-channel images for both indoor and outdoor datasets and their annotations can be seen in Figure 7.2.

The outdoor dataset is built from the 477 RGB and NIR image pairs released by Brown and Süsstrunk [2011]. From these images, we discard 107 images due to mis-registration and ambiguity of classes. The rest of the images are manually labeled at the pixel level, thus yielding pixel segmentation masks. The labels are selected from 10 predefined classes[2]:

[2] In parentheses we list the number of images containing the given class.

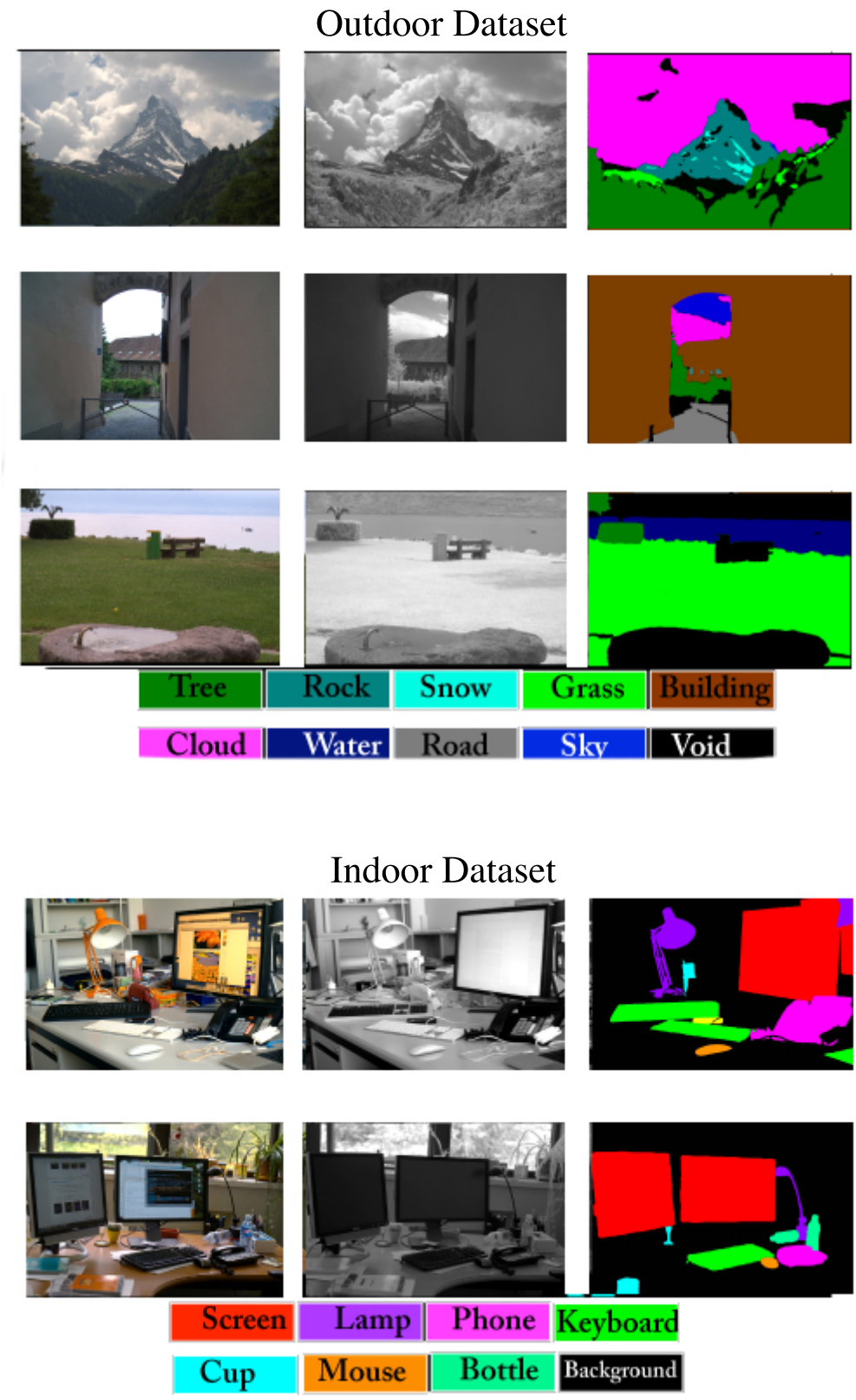


Figure 7.2: Sample images from our outdoor and indoor datasets: RGB (left), NIR (middle). ground truth(right).

Building (179), Cloud (161), Grass (159), Road (108), Rock (80), Sky (174), Snow (41), Soil (78), Tree (274), and Water (79). We follow the MSRC dataset's annotation style [Shotton et al., 2006], i.e., each pixel is labeled as belonging to one of the above classes or to a Void class. The latter corresponds to pixels whose class is not defined as part of our classes of interest, or which are are too ambiguous to be labeled. Similarly to Shotton et al. [2006], we discard pixels of the Void class for the evaluation.

The indoor dataset consists of 400 images we gathered ourselves in various office environments. The registration between RGB and NIR images is conducted, as with the outdoor dataset [Brown and Süsstrunk, 2011], by using the algorithm proposed in Szeliski [2006]. The images are processed using automatic max-RGB white balancing for the RGB components, and equal weights on the RGB sensor responses for the NIR components, as explained in Fredembach and Süsstrunk [2008]. For these images, we select 12 object categories: Screen (206), Clothing items (184), Keyboard (178), Cellphone (108), Mouse (145), Office phone (113), Cup (163), Bottle (130), Potted plant (77), Bag (123), Office lamp (70), Can (59). These objects are manually segmented and annotated at the pixel level, as in the Pascal VOC Challenge [Everingham et al., a], where all pixels not belonging to the predefined classes are considered as Background (the 13th class). Contrary to the Void class, the segmentation performance of the predicted background is evaluated.

However, the Background class is rather diverse. Therefore, instead of modeling it explicitly, we first predict only the other classes and then we employe a minimum level of confidence threshold on the predicted classification scores. If the maximum posterior probability is smaller than a single universal threshold (in our case $\mathscr{T} = 0.5$), the pixel is labeled as Background, otherwise it is labeled with the class label for which the maximum was found. In other words, in our CRF model, given the observation $\mathbf{D}$, $P(X = Background \mid \mathbf{D}) = \mathscr{T}$.

7.2.1 Evaluation Procedure

In all our experiments, we randomly split the dataset into 5 sets of images (5 folds) and define 5 sets of experiments accordingly. For each experiment, one fold is used as the testing set and the remaining images are used for training the model. Results (predicted segmentation maps) for the 5 test-folds are grouped all together and evaluated at once, producing a single score for each evaluation measure.

To compare the segmentation results of different methods and parameters, we use both region-based and contour-based measures. The latter are mainly to evaluate the segmentation results

with a criterium that focuses on the quality of the segmentation borders.

Region based accuracies are generally based on the confusion matrix **C** that can be computed either individually for each image or as an aggregation of predictions (cumulating the predictions) for the whole dataset as follows:

$$\mathbf{C}_{kl} = \sum_I |\{p_i \in I | S^I_{gt}(p_i) = k \,\&\, S^I_{pr}(p_i) = l\}|$$

where S^I_{gt} is the ground truth segmentation map of the image I, S^I_{pr} the predicted segmentation map and $|A|$ is the number of elements in the set. Hence $\mathbf{C}_{kl}$ (the kl^{th} element of matrix **C**) represents the number of pixels with ground-truth class label $k \in \mathscr{L}$, which is predicted with the label $l \in \mathscr{L}$.

Denoting by $\mathbf{G}_k = \sum_l \mathbf{C}_{kl}$, the total number of ground-truth pixels labeled with k, and by $\mathbf{P}_l = \sum_k \mathbf{C}_{kl}$ the total number of predicted pixels labeled with l, we can define the following evaluation measures:

- *Overall Pixel Accuracy (OA)* measures the ratio of correctly labeled pixels:

$$OA = \frac{\sum_{k=l_1}^{l_n} \mathbf{C}_{kk}}{\sum_{k=l_1}^{l_n} \mathbf{G}_k}$$

- *Per Class Accuracy (CA)* measures the ratio of correctly labeled pixels for each class and then averages over all classes:

$$CA = \frac{1}{|\mathscr{L}|} \sum_{k=l_1}^{l_n} \frac{\mathbf{C}_{kk}}{\mathbf{G}_k}$$

- *The Jaccard Index (JI)* measures the intersection over the union of the labeled segments. This measure is computed by dividing the diagonal value $\mathbf{C}_{ii}$ (true positives) by the sum of all false positives and all false negatives for a given class $k \in \mathscr{L}$:

$$JI = \frac{1}{|\mathscr{L}|} \sum_{k=l_1}^{l_n} \frac{\mathbf{C}_{kk}}{\mathbf{G}_k + \mathbf{P}_k - \mathbf{C}_{kk}}$$

Note that OA and CA correspond to the measures used in general to compare segmentation results on the MSRC dataset [Shotton et al., 2006], whereas JI is the measure used in the Pascal

Segmentation Challenge [Everingham et al., a].

The trimap accuracy (TrimapAcc) evaluates segmentation accuracy around boundaries [Kohli et al., 2009]. The idea of this measure is to build a narrow-band around each contour and to compute pixel accuracies OA evaluating only the pixels within the given band. As a single band gives only partial evaulation, the size of this band r is varied and the overall accuracy values (denoted here by $T(r)$) is plotted as a curve. This allows a visual comparison of 2 or more methods.

Statistical significance tests To examine if our results are statistically different, we also compute paired t-test on the set of image based results. In this case we compute results per image and compare the two-score distributions. The paired t-test computes the probability p-value of the hypothesis $H1$, the two distributions have the same mean. Accordingly, a p-value< 0.05 results in the rejection of $H1$ (same means) at the 95% confidence level. In such cases, we can say that the two methods generating the respective mean results are significantly different.

7.3 Experimental Results

In Section 7.1, we describe different ways to integrate NIR information into our segmentation framework. In this section, we investigate and compare these different issues through a set of experiments. First, we consider only the recognition part ($\lambda = 0$, hence only the unary term) and compare different descriptors and combinations of visible and NIR based features. The study for the regularization part (adding the pairwise energy term) is conducted only for the best performing recognition models.

7.3.1 The Recognition Part

To compare different descriptors and to evaluate them in the recognition part, we produce a semantic segmentation by assigning pixels to their most likely label with

$$\mathbf{x}^* = \text{argmax}_{\mathbf{x}\in\mathscr{L}^C} \sum_{i\in v} P(X_i = \mathbf{x}_i \mid \mathbf{D})$$

given the observation $\mathbf{D}$. This is equivalent to the full model when using $\lambda = 0$.

As mentioned before in section 7.1.1, although other features could have been considered, here we will focus on two type of features: SIFT [Lowe, 2004] and local color statistics [Clinchant

Method	Outdoor			Indoor		
Descriptor	**CA**	**OA**	**JI**	**CA**	**OA**	**JI**
COL_{rgb}	74.07	78.25	59.21	39.94	50.49	24.23
COL_{rgbn}	76.18	80.56	61.94	45.33	56.03	28.74
COL_{p1234}	76.95	80.63	63.00	44.57	54.27	28.10
$SIFT_l$	66.88	73.36	50.68	49.19	47.55	32.49
$SIFT_n$	67.07	73.96	51.12	48.32	43.46	31.54
$SIFT_{p1}$	61.01	73.44	50.91	48.30	44.02	31.22
$SIFT_{rgb}$	75.07	80.13	60.33	49.75	51.91	33.03
$SIFT_{rgbn}$	76.47	82.38	62.41	**53.79**	58.98	**36.77**
$SIFT_{p1234}$	76.74	82.55	62.77	49.75	57.41	33.09
$COL_{rgb} + SIFT_l$	79.17	83.52	65.85	49.47	56.50	32.36
$COL_{rgbn} + SIFT_l$	80.18	84.76	67.34	53.64	60.95	36.26
$COL_{rgbn} + SIFT_n$	80.13	84.88	67.40	53.20	**61.13**	35.78
$COL_{p1234} + SIFT_n$	**80.91**	**85.19**	**68.46**	52.78	60.23	35.44

Table 7.2: Evaluation (average of per-class, overall accuracies, and Jaccard index) of the segmentation for different local descriptors and their combinations both on outdoor and indoor datasets.

et al., 2007], denoted by COL. To compute these features in any of the considered channels (R,G,B,N, luma L or alternative color spaces P1,P2,P3,P4), we proceed as follows.

We extract 32×32-sized patches on a regular grid (every 10 pixels), at 5 different scales (the first 5 terms of the geometric series with ratio $\sqrt{2}$) in the given channel. Hence the coarsest scale corresponds to resizing the patch by a factor of 4 ($\sqrt{2}^4$) and the finest corresponds to the un-resized patch ($\sqrt{2}^0$).

Low-level descriptors are computed for each patch. We consider two different descriptors: $SIFT$, which is 128-dimensional as we compute orientation histograms of 8 bin using a 4x4 grid on the patch; and COL, which is 32-dimensional as we consider the mean and the variance of the color intensity using the same 4x4 grid of the patch. In general, we consider more than one channel at a time (at least 3) by concatenating the corresponding features, hence COL_{rgb} will be 96-dimensional, COL_{rgbn} 128-dimensional and $SIFT_{rgbn}$ 512-dimensional. For a fair comparison, we reduce all features to 96 dimensions by using PCA. In the projected space, a GMM-based visual codebook of 128 Gaussians is built and used to transform the low-level features into a FV.

By using the same PCA dimension and the same codebook size, the FV representation of all descriptors share the same dimension. Sparse logistic regression (SLR) [Krishnapuram et al.,

2005] is used for classification and the ultimate output is the $\frac{1}{1+\exp(-s_k)}$, where $s_k, k \in \mathscr{L}$ are the scores of the FVs according to their class relevances. The class probabilities of each pixel $P(X_i = k \mid \mathbf{D})$ are then computed as a weighted average of the patch posteriors as described in Section 7.1.1. For each pixel, the label corresponding to the highest score is retained yielding a predicted segmentation map. In the case of the indoor datasets, this score is further compared to the threshold $\mathscr{T} = 0.5$ and if the highest score is below this threshold, the pixel is assigned to the Background class. The accuracy of the predicted segmentations are then evaluated with different region-based accuracy measures described in Section 7.2.1. Note that here the method does not seek to follow any object regions and there is no regularization applied, therefore using contour based evaluation measures to compare these methods does not make much sense.

In Table 7.2 (upper part), we show the segmentation results obtained with region-based accuracy measures for different local descriptors. From these tables we can observe that the accuracy of the recognition using *COL* features is significantly higher when NIR descriptors are considered in conjunction with RGB (COL_{rgbn}, COL_{p1234}), compared to the visible-only scenario (COL_{rgb}). Similarly, when we combine several SIFT features, $SIFT_{rgbn}$ and $SIFT_{p1234}$ outperform $SIFT_{rgbn}$ containing only features from the visible image.

$SIFT_n$ performs similarly to $SIFT_l$, leading to slightly better performance in outdoor environment, and slightly worse for the indoor dataset. The reason might be that, in the NIR image, material-intrinsic texture properties are captured, which might be insufficient to describe the appearance of our objects in the indoor dataset. For most classes in the outdoor dataset, however, this appearance seems to be better captured in N than in the L channel.

In both cases, best results are obtained with multi-spectral SIFT when both visible and NIR image is considered. Note however, that these features incorporate both texture (explicitly) but also color (implicitly, considering the SIFT in multiple channel). Another way to combine color and texture is by early or late fusion of *COL* and *SIFT*. As in Salamati et al. [2011b] it is clearly shown for image categorization that the late fusion of these features outperforms early fusion; here we do not consider the latter. The results of late fusion between different *COL* and *SIFT* are shown in Table 7.2 (lower part). Note that $COL_{rgb} + SIFT_l$ corresponds to our visible baseline for the recognition part[3].

Comparing the results of late fusion of *COL* and *SIFT* to the multi-spectral SIFT, we can observe the following: In the case of the outdoor dataset, the late fusion of *COL* and *SIFT*

[3] Note that it is similar to the approach of Csurka and Perronnin [2011] without region labeling and without global score-based fast rejection.

Method		Outdoor			Indoor		
Descriptor	**Pairwise**	**CA**	**OA**	**JI**	**CA**	**OA**	**JI**
$SIFT_{rgb}$	VIS	77.02	82.40	62.90	51.60	60.94	34.69
	NIR	77.00	82.15	62.86	51.47	60.98	33.42
	$VIS+NIR$	77.05	82.40	62.91	51.75	60.86	34.79
$SIFT_{rgbn}$	VIS	78.07	84.02	64.54	55.3	68	38.5
	NIR	78.07	83.99	64.54	55.16	68	38.44
	$VIS+NIR$	78.25	84.17	64.80	**55.67**	68.02	**38.86**
$SIFT_{p1234}$	VIS	78.30	84.22	67.97	50.72	65.82	34.10
	NIR	78.30	84.11	67.97	51.27	66.00	34.67
	$VIS+NIR$	78.35	84.27	68.04	51.26	65.85	34.59
$COL_{rgb}+SIFT_l$	VIS	79.97	84.56	67.14	50.84	63.58	33.61
	NIR	80.05	84.73	67.06	50.70	63.81	33.46
	$VIS+NIR$	80.24	84.87	67.40	51.113	63.57	33.87
$COL_{rgbn}+SIFT_l$	VIS	81.22	85.90	68.82	54.54	68.23	37.06
	NIR	81.15	85.88	68.78	54.63	68.24	37.15
	$VIS+NIR$	81.22	85.97	68.87	54.65	68.27	37.11
$COL_{rgbn}+SIFT_n$	VIS	81.14	86.08	68.89	53.86	68.78	36.37
	NIR	81.20	86.07	69.02	53.97	68.75	36.48
	$VIS+NIR$	81.31	86.22	69.15	54.37	**68.78**	36.89
$COL_{p1234}+SIFT_n$	VIS	81.69	86.22	69.61	53.54	67.97	36.32
	NIR	81.56	86.01	69.47	53.77	67.93	36.05
	$VIS+NIR$	**81.86**	**86.34**	**69.86**	53.91	67.87	36.37

Table 7.3: Results for the full CRF model both for outdoor and indoor datasets.

clearly outperforms the multi-spectral SIFT, whereas this is less true in the case of the indoor dataset where the two strategies yield similar results. The main reason is probably that color (RGB) is much more important in the case of the outdoor dataset that in the indoor dataset where most objects can have different colors that are not specific to a given class. This observation is confirmed by the low performance of COL_{rgb} compared to the $SIFT_l$ in the case of indoor, while for the outdoor dataset COL_{rgb} significantly outperforms $SIFT_l$, showing how important the color is for predicting the appearance of these scene classes (e.g., Sky, Grass, Snow, etc). Note that adding the NIR information is very helpful because the intensity in the N channel captures material information and is more informative concerning these object classes.

Although the best strategy is dataset dependent, $COL_{rgbn}+SIFT_n$ seems to be a good allround choice if only a single strategy has to be selected.

7.3.2 The Full CRF Model

In this section, we consider the most promising recognition models, both for visible only and for the visible+NIR images, and we apply the full CRF model with regularization based on the RGB image (VIS), on the NIR information (NIR) and on both ($VIS+NIR$). In our model (Equation 7.1), we used a fixed weight parameter $\lambda = 5$ in all our experiments. Results are shown in Table 7.3 from which we can deduce the following conclusions:

First, we can see that, consistently for all the descriptors, the accuracy of semantic segmentation increases when we consider both visible and NIR channels ($VIS+NIR$) in the pairwise potential compared to using only visual or only NIR. Second, as expected, the regularization (any of them) improves the segmentation results obtained for any recognition model compared to the segmentation without regularization ($\lambda = 0$). Finally, the ranking between the models with different descriptors is similar with or without regularization given a regularization model (e.g., VIS or $VIS+NIR$). This is again not surprising, as we use the same regularization term that is independent of the feature used in the recognition part. Hence again $COL_{rgbn} + SIFT_n$ and $COL_{p1234} + SIFT_n$ lead to the best performances in the case of the outdoor dataset, and $SIFT_{rgbn}$ performs best in the case of the indoor dataset, $COL_{rgbn} + SIFT_n$ being second best. Compared to the visible-only baselines $COL_{rgb} + SIFT_l$ or $SIFT_{rgb}$ with visible image-based regularization (VIS), they are significantly better and statistically different at the 95% confidence level according to our t-test applied to the score distributions.

In this section, we are mainly interested in the gain we have when we compare the VIS or NIR-based regularization with the $VIS+NIR$-based regularization. Considering region based evaluation measures, we can see only slight improvements even if the paired t-test often shows significant differences between the corresponding score distributions. Therefore, to better evaluate the gain by adding NIR to VIS in the regularization, we also evaluate some of these results with contour based measures.

We apply the trimap accuracy [Kohli et al., 2009] (overall pixel accuracy in the neighborhood of object boundaries varying the boundary size) and show some of the results in Figure 7.3 and 7.4.

From these results, we can first notice the importance of the regularization. Indeed in both cases, any of the edge potentials we use as regularization leads to a significant improvement (statistically different at the 95% confidence level according to our t-test) on the results obtained by recognition alone (NO, for no pairwise). Comparing different edge potentials, we can see that, in outdoor scenes, using the visible image leads to better segmentation than

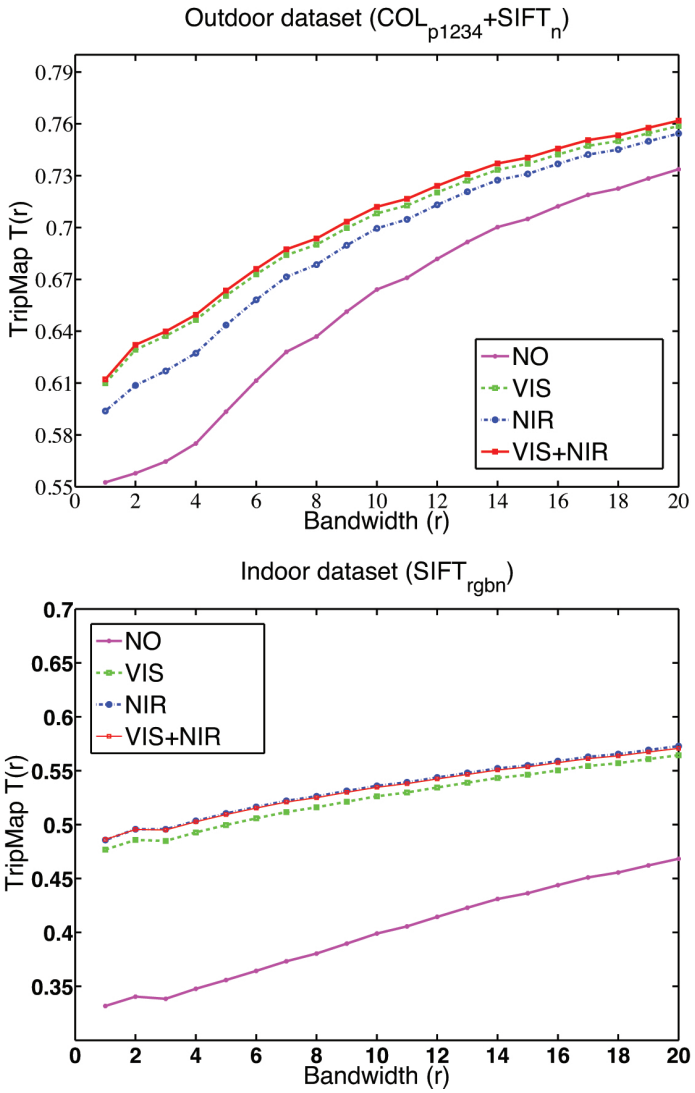


Figure 7.3: The TrimapAcc plots with different pairwise potentials using $COL_{p1234} + SIFT_n$ (top- for the outdoor dataset) respectively $SIFT_{rgbn}$ (bottom-for the indoor dataset) as unary potential.

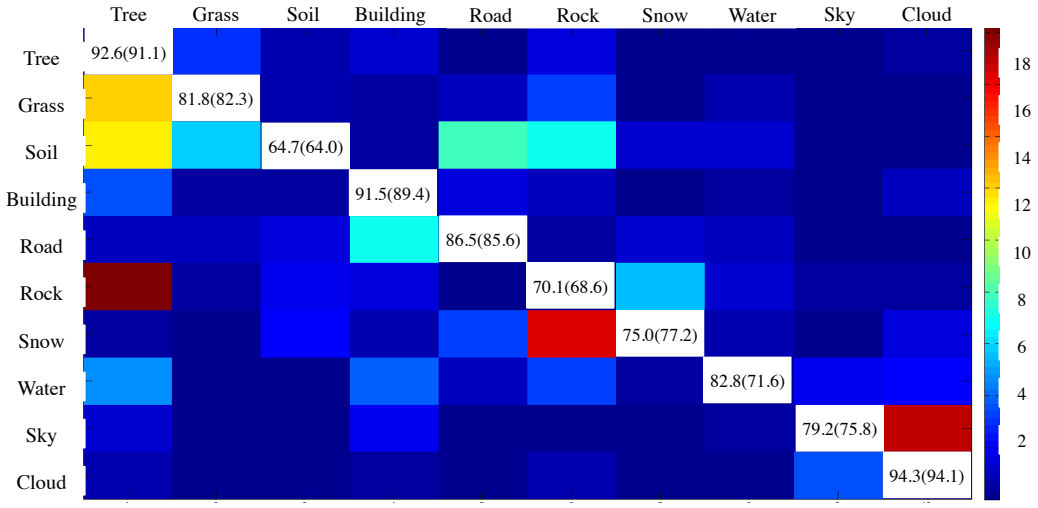


Table 7.4: Confusion matrix of $COL_{p1234} + SIFT_n$ and four-dimensional pairwise. For each class, the corresponding segmentation rates for the best visible scenario ($COL_{rgb} + SIFT_l$ with visible-only pairwise) are given in parentheses. Outdoor dataset.

using the NIR image alone[4]. Fixing the bandwidth at 5 pixels and running the t-test, we found that all $T(5)$ are statistically different at 95% confidence level.

In order to show also some qualitative comparisons, in Figure 7.9 we further show a few segmentation results obtained with the visible baseline and the best visible + NIR setting. These images show again that incorporating NIR significantly increases the border accuracy.

[4]This behavior can be partially explained by the fact that the manual annotation was done in the RGB images and in some images of outdoor dataset, the position of some objects, such as clouds or cars, are different between the two representations because visible and NIR images were acquired in two consecutive shots. Whereas, in the indoor scenes, there is no significant movement between two shots, as no moving objects are present.

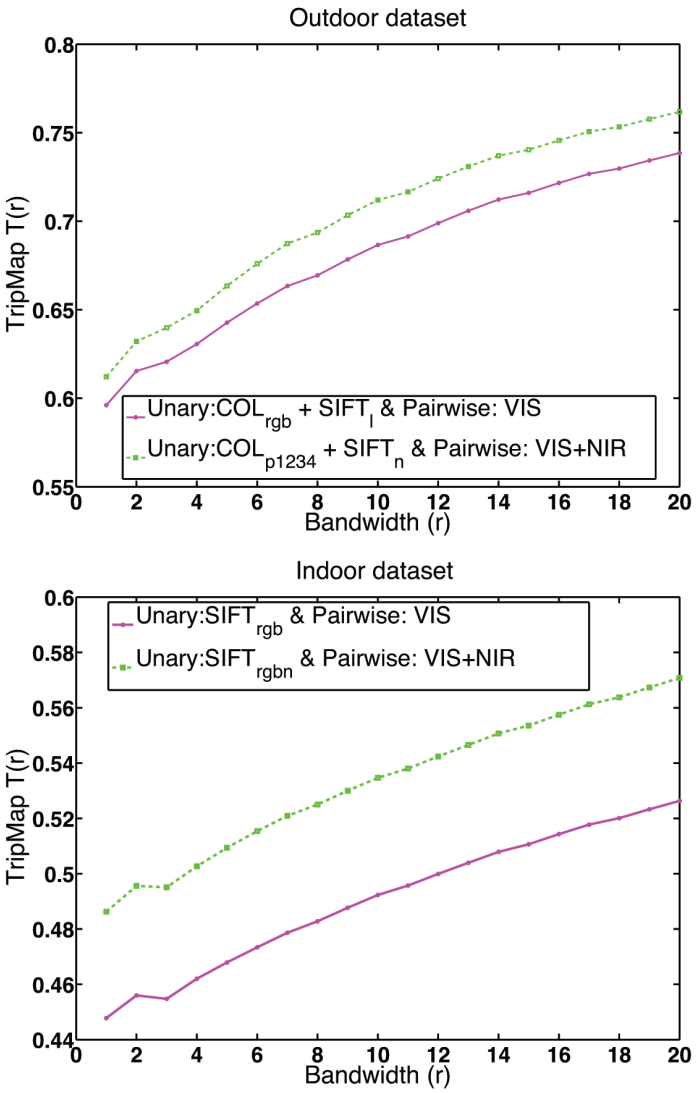


Figure 7.4: The TrimapAcc plots compare the border accuracy of the results of the visible only scenario and the proposed strategy, top-for the outdoor dataset and bottom-for the indoor dataset.

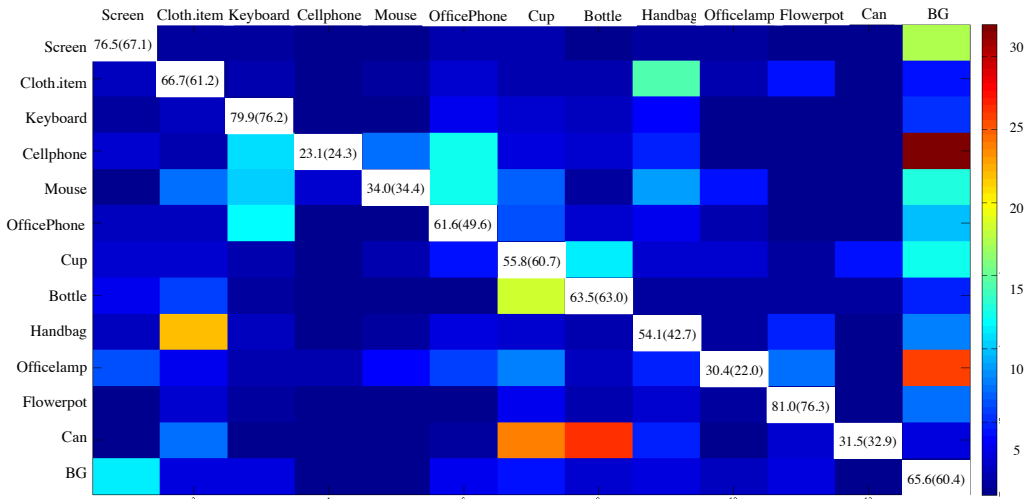


Table 7.5: Confusion matrix of $SIFT_{rgbn}$ and four-dimensional pairwise. For each class, the corresponding segmentation rates for the best visible scenario ($SIFT_{rgb}$ with visible-only pairwise) are given in the parenthesis. Indoor dataset.

7.4 Class-Based Analysis and Discussion

In this section, we analyze and compare class-by-class the segmentation results obtained with the best visible baseline and the best NIR integration strategy. To do this, we show confusion matrix Tables 7.4 and 7.5 and example segmentations in Figure 7.9.

From the results, we can deduce the following observations:

Haze Effect. The benefit of using the NIR channel in the presence of haze can be observed particularly in the case of Sky, Tree and Rock classes, the latter often representing mountains. As stated by Rayleigh's law, the light scattered from very small particles ($< \lambda/10$) is inversely proportional to the fourth power of the wavelength λ (i.e., $\propto 1/\lambda^4$) [Fredembach and Süsstrunk, 2008]. Particles in the air (haze) satisfy this condition and are scattered more in the short-wavelength range of the spectrum. Thus, when images are captured in the NIR, atmospheric haze is less visible and the sky becomes darker (see Figure 7.5). The "haze transparency" characteristic of NIR results in sharper images for distant objects. In particular, vegetation at a distance in the visible image is smoothed and bluish, which can affect the performance of texture and color features in the classification task. The sharp and haze-free appearance of vegetation in NIR images helps classification and leads to better segmentation (see also the diagonal in Table 7.4).

Border Accuracy. In both datasets, for many images, we observe that borders are more

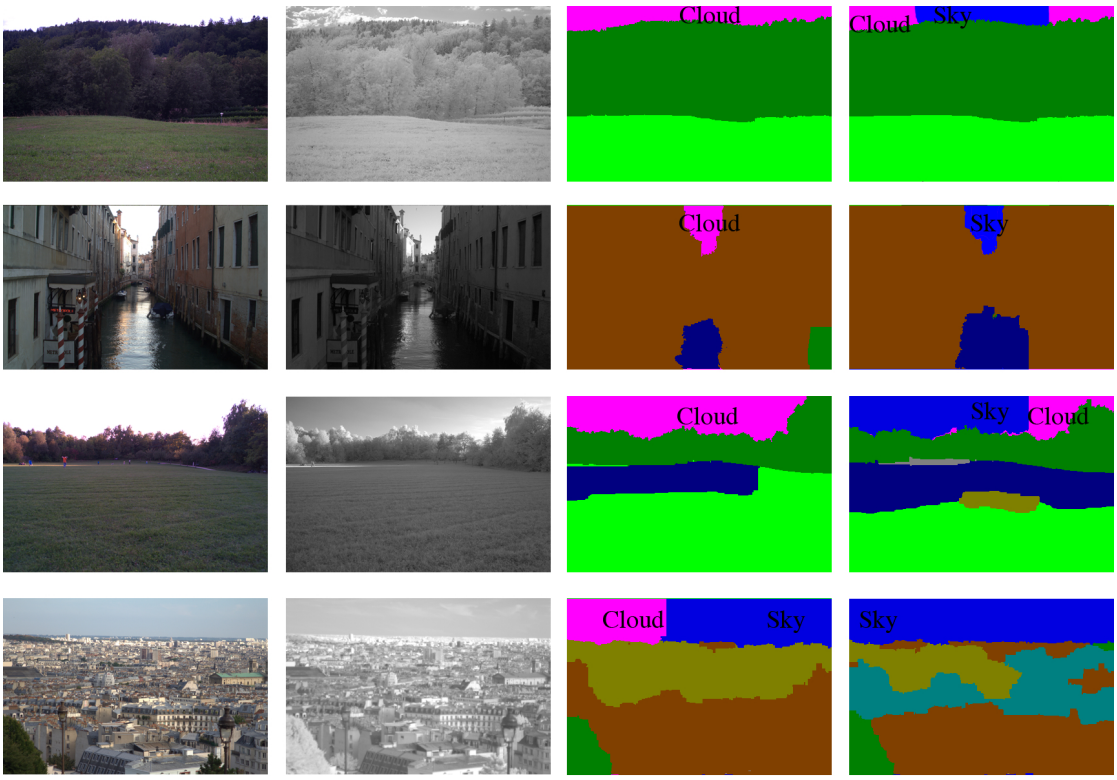


Figure 7.5: Examples from the outdoor dataset. Note the better classification and recognition of Clouds and Sky when NIR information is incorporated.

precisely detected when NIR information is incorporated in the pairwise potential. This can be explained by the material dependency of NIR responses that may reduce wrong edges due to clutter, or may result in more contrasted edges between classes. This information, used in the regularization part of our model, helps us to better align borders between regions with the material change (see Figure 7.6).

The Relevance of Colors. Classes in the outdoor dataset are better recognized by their intrinsic color such as Sky, Grass, Water and Cloud. Capturing texture in a one-channel image and fusing it with the color information improves the results, mostly by distinguishing between Grass and Tree, or Sky and Water, where color is less discriminative. Figure 7.7 shows that incorporating material-dependent NIR images in the COL descriptor and fusing it with $SIFT$ features on the NIR image helps to recognize such confusing classes. A linear conversion in the color space from RGBN to PCA for the COL descriptor improves the accuracy of results.

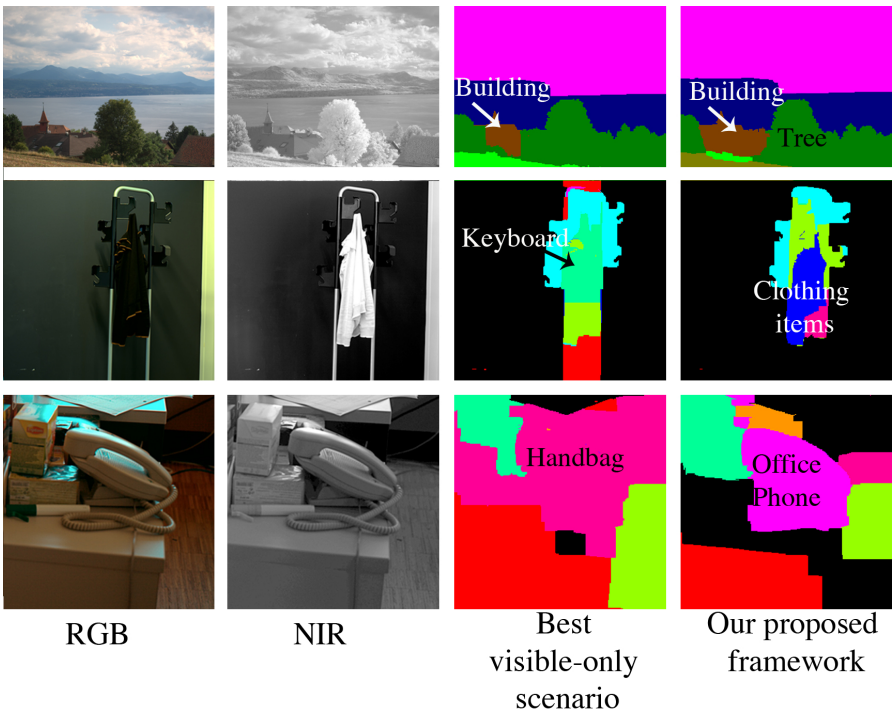


Figure 7.6: Examples from both outdoor and indoor datasets. Note that the material dependency of NIR images results in more accurate detection of object boundaries.

By contrast, in the indoor scenes, most of the classes are man-made, hence they contain many colors (i.e., color is less distinctive for recognizing the classes). For example, Clothing items can be colored in various different colors and patterns, but the texture of textile material is pretty unique and discriminative. This is also the case for Screen and Handbag. This can explain why COL features perform poorly and hence multi-spectral SIFT outperforms the late fusion of COL and $SIFT$ features. Figure 7.8 shows that incorporating COL gives poor results in the recognition of colorful classes such as Screen, Clothing items, and Handbag. In such classes, texture is more intrinsic to the class, therefore multispectral-SIFT ($SIFT_{rgbn}$) outperforms the late fusion of COL and $SIFT$.

De-correlated Space. Overall, going to the PCA-based de-correlated space gives better performance for the outdoor dataset than for the indoor dataset, where the performance is in general decreased.

Material Relevance. In general, the classes that correspond to a specific type of material, such as Water, Tree, Sky, Cloth, Screen, Officephone (plastic), Handbag (fabric, suede, leather), and Flowerpot exhibit the largest improvement (up to 17%) when NIR is added. In this cases, even

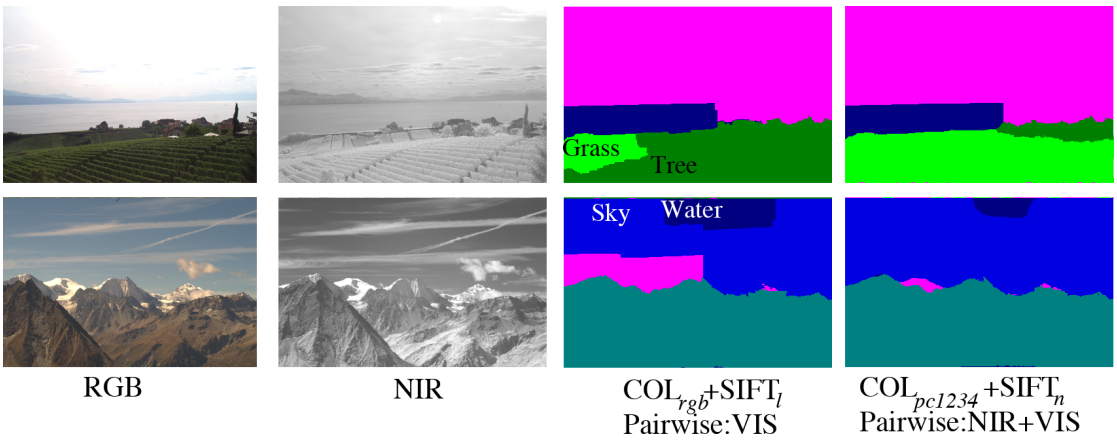


Figure 7.7: The material dependency characteristics of NIR images helps to distinguish more accurately between the classes of material with the same intrinsic color. Higher contrast in the NIR images in the sky makes $SIFT_n$ a more discriminative feature in distinguishing between Sky and Water.

if in visible images the color can be confused with other classes (Sky, far away Mountains), they have a unique four-dimensional appearance that leads to a significant improvement compared to the visible baseline (see e.g., (d) and (e) in Figure 7.2).

7.5 Incorporating Shadow Information in the CRF Model

A common problem in computer vision, when dealing with color images, is the presence of shadows. A shadow is cast when an object occludes a light source. Due to the difference between the light intensity reaching a shaded region and a directly lit region, shadows often are characterised by conspicuously strong brightness gradients. These physical effects cause strong edges in an image and can be detected as an object edge by our proposed segmentation technique. Commonly, the edges caused by shadows are not considered for a human when segmenting, but the shadows cast would be detected by current segmentation algorithms (Figure 7.10 shows an example of this). It is therefore of great interest to discover ways of properly detecting shadow edges and removing them from the final result. Fredembach and Süsstrunk [2010] presents a simple though accurate shadow detection method that employs the features offered by the NIR band, along with color information. Using the property of the NIR band, in which most of the colorants are transparent or have higher reflectance [16], the authors show that combining the dark map of both visible and NIR images with ratios of the color channels (red, green and blue) to NIR identifies the pixels that are shadow candidates.

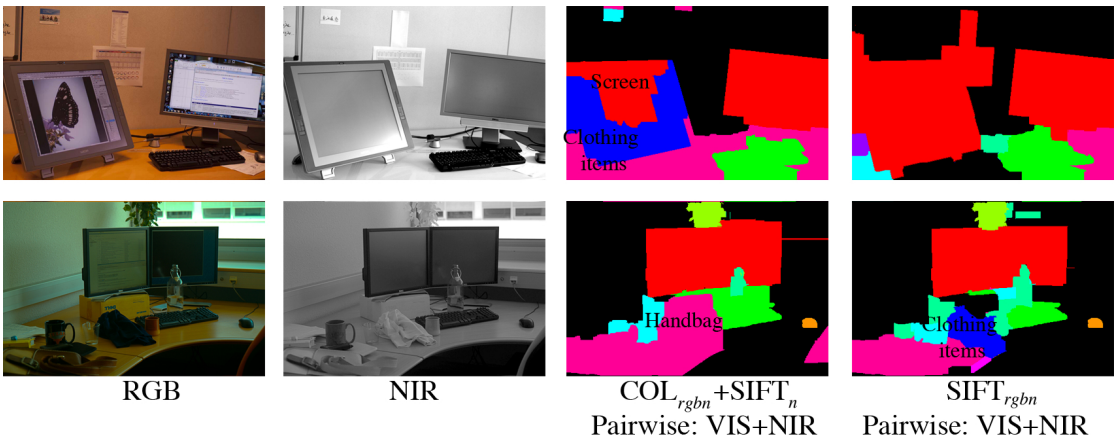


Figure 7.8: multispectral-SIFT ($SIFT_{rgbn}$) outperforms the late fusion of COL and $SIFT$ in recognition of colorful classes where texture is more intrinsic to the class.

Implementing the method by Fredembach and Süsstrunk [2010], we can form a shadow probability map that gives the probability of each pixel being under the shadow. Salamati et al. [2011a] suggest incorporating the shadow probability map and forming a shadow-free image by removing the shadow edges whose surroundings have similar chromaticity values (See Appendix B for more details).

In this section, we evaluate the possibility of using a shadow map of an RGB and NIR image to perform a segmentation that is robust to shadows.

Given the RGB and NIR representations of a scene, a high-quality shadow map can be computed by Fredembach and Süsstrunk [2010]. Their proposed method is based on the fact that shadows are generally found in the dark parts of an image, be it color or NIR. By observing that commonly encountered light sources have very distinct spectra in the NIR, they proposed that the ratios of the color channels (red, green and blue) to the NIR give valuable information about impinging illumination, which they employed to assess the shadow candidate pixels.

The process of how to find shadow-candidate pixels M (Equation B.3) is fully discussed in Appendix B. To obtain the final shadow mask, M needs to be binarized. To this end, Fredembach and Süsstrunk [2010] proposes to compute the histogram of M and calculate the location of its first valley. Let us denote this location as θ. The binary shadow value of each

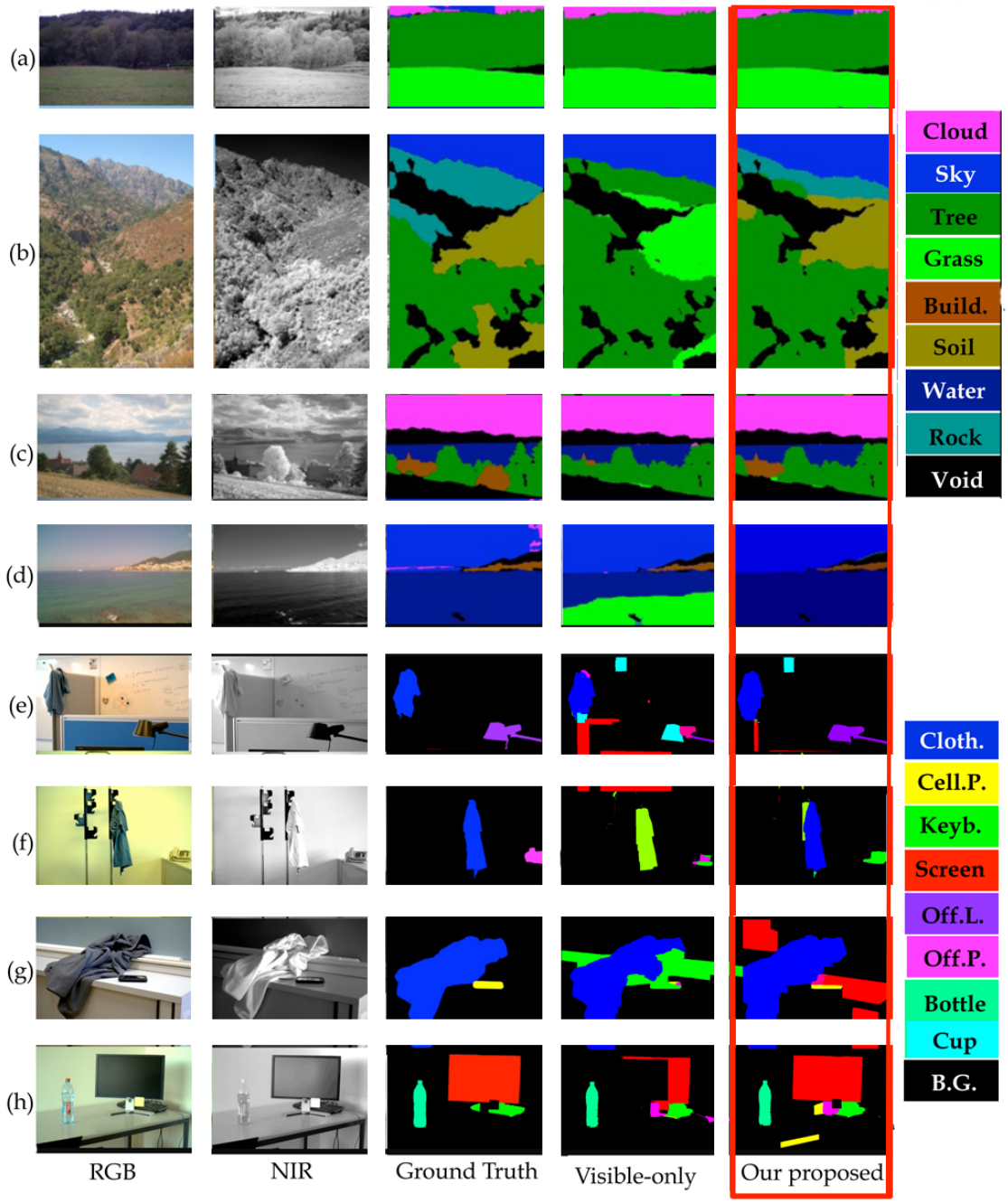


Figure 7.9: Sample segmentation results for the outdoor and indoor datasets.

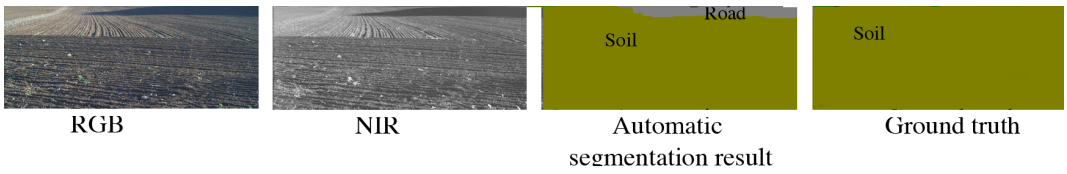


Figure 7.10: A scene in both visible and NIR representations with a strong cast shadow. The cast shadow is mis-labeled by our best segmentation algorithms.

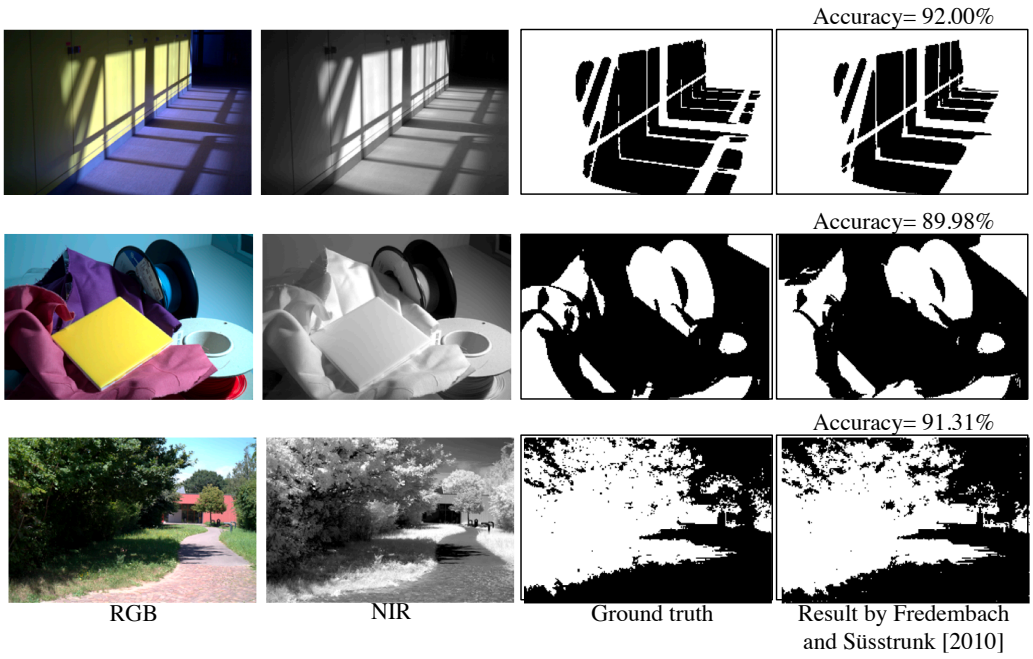


Figure 7.11: Input images (RGB and NIR, the manually labelled ground truth, as well as the resulting shadow masks by Fredembach and Süsstrunk [2010].

pixel x is then given by:

$$M_{bin}(x) = \begin{cases} 1 & \text{if } M(x) < \theta \\ 0 & \text{otherwise} \end{cases} \tag{7.4}$$

Some results of this method are shown in Figure 7.11. The shadow detection accuracy of

this framework is 89%(±8). To distinguish between shadow edges and object boundaries, we propose to construct our pairwise potential in the CRF model, and modify the pairwise potential to assign a smaller penalty when crossing the shadow edges.

$$E_{pair}(\mathbf{x}) = \sum_{(i,j)\in v} (1-\delta_{x_i,x_j}) \exp(-\beta \| p_i - p_j \|^2) \cdot (1 - \eta \operatorname{xor}(M_{bin}(x_i), M_{bin}(x_j))) \tag{7.5}$$

We evaluate this model on a saliency-based object segmentation scenario. In this scenario, we assume that we are looking for the most salient object of the entire scene, and that we want to segment it from the background. For this purpose, we assume that a saliency extraction method is available. Any method could be used. In our experiments, we use a color based saliency method [Achanta et al., 2009]. We use the saliency method as a coarse localization method in the 4-channel data, by plugging it into the unary term of our energy function. Given the saliency map (Ξ), we now assume that for each pixel, we know its probability to belong to the object of interest. This is the information we use in our graph. More precisely, the saliency value of the pixels can be encoded with a probability map over the pixels, and can be included in the unary term as

$$E_{un}(\mathbf{x}) = \sum_{i\in v} -\log(\Xi(x_i)) \tag{7.6}$$

Some results are presented in Figure 7.12, for all the samples $\eta = 1$. For these examples, the CRF model fails to detect the boundary of the actual physical object, because of the large difference in pixel values around the shadow edges. Using the shadow map and modified pairwise potential in our model, the object boundaries are detected more accurately.

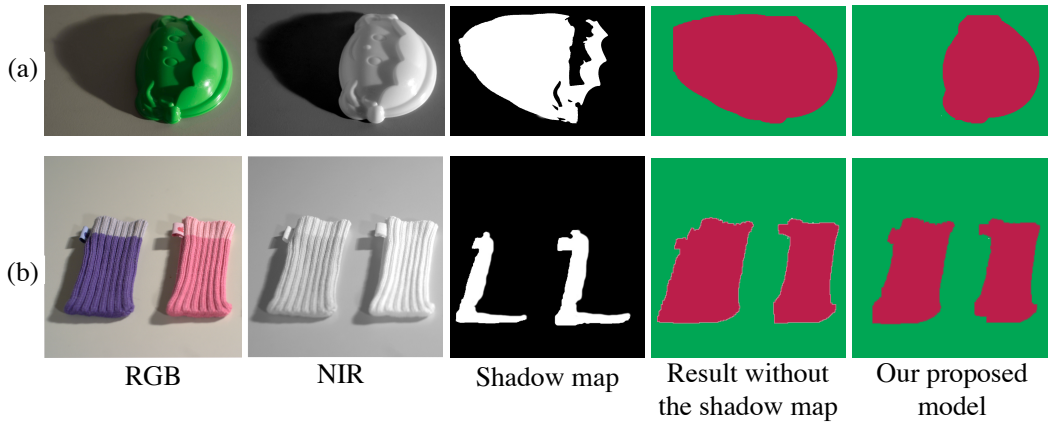


Figure 7.12: Binary segmentation of images with strong presence of shadows. Incorporating shadow mask in the pairwise potential increases the precision of the result by 10% in image (a) and 1% in image (b).

8 Conclusion and Future Work

We have explored the idea that near-infrared (NIR) information, captured from an ordinary digital camera, could be useful in scene understanding and visual recognition. We have shown in this thesis how, unlike remote sensing and military communities (that employ expensive hardware to record NIR and experts to digest this information), NIR can be useful in everyday photography. We have shown that the NIR channel has correlations with the RGB channels that are significantly lower than those of RGB to each other, and we have studied how to effectively exploit these differences to improve scene recognition and semantic segmentation performance. In this chapter, we provide a summary of the major contributions and findings of the work presented in this dissertation. We also outline prospective directions for further research.

- **Human Cognition:** We have conducted a user study that shows the usefulness of using NIR information in scene understanding by humans. We have shown that, in most scenarios, the cognition threshold for NIR images is significantly lower than that of the visible images. One of the basic differences between NIR and visible images is that edges in the NIR image correspond mostly to physical object boundaries, and the results are also consistent with the hypothesis that the essential features for visual cognition are object boundaries rather than color and texture.

  It would be interesting to study the human cognition threshold in a scene representation that only contains the edges of the original scene. This "structural-based" representation for scene recognition is studied in [Rouse and Hemami, 2007], where the images represent the scene in the visible part of the spectrum. As future work, we could examine the performance of NIR and visible structural-based representations and analyze the advantages and disadvantages of the structural representations of NIR images compared

to the visible images.

- **Image Classification:** By extending the result of incorporating NIR images in cognition tasks to automatic scene classification, we have shown that NIR is useful information that, combined with visible cues, can improve this task. We have also investigated the best way to include the 4-dimensional color information in our categorisation method. This study is based on the Fisher Vector (FV) representation, a generic and powerful categorisation framework. As a conclusion specific to this method, we have shown the usefulness of applying a PCA projection to local descriptors. We have proposed using a color descriptor that encodes local statistics about color information and the NIR channel, and we have shown that the best results can be achieved by the late fusion of FV signatures computed on the best color descriptor and on the NIR SIFT descriptor. Our proposed framework outperforms the state-of-the-art framework by 15%.

- **Material-Based Boundary Detection:** We have presented a method that accurately detects physical object boundaries in images by using RGB and NIR information. By incorporating NIR as well as RGB channels, we have extended the 4-sensor camera calibration model to represent images invariant to shading and shadows. The changes in color within an object confound many segmentation algorithms so that they assign different segments to parts of the object in different colors. Due to the transparency of most of the colorants in the NIR part of the spectrum, the edges in an NIR image correspond mostly to the boundary of objects in the scene. Hence, in NIR images, low-level segmentation frameworks do not confuse object boundaries with the color patterns of the object. Combining the segmentation results of the shadow-free images with those of NIR images, we have proposed a framework that leads to segments that are based on changes in material.

- **Semantic Image Segmentation:** We have also presented a framework for semantic image segmentation using RGB and NIR information. We propose to formulate the segmentation problem by using a CRF model, and we have studied how to incorporate the NIR cue, either in the recognition part or in the regularization part of our model. Considering the characteristics of NIR images, we have defined color and SIFT features on different combinations of the RGB and NIR channels. To evaluate this framework, we have introduced a novel database of outdoor and indoor scene images, annotated at the pixel level, with 10 categories in the outdoor and 13 categories in the indoor scenes.

  We have shown that integrating NIR as additional information along with conventional RGB images improves the segmentation results. In particular, the overall improvement

is due to a large improvement for certain classes whose response in the NIR domain is particularly discriminant, such as Water, Sky or Screen. One of our contributions is that we systematically studied the reason for this improvement taking into consideration the material characteristics and the properties of the NIR wavelength range.

To achieve even more accurate results that are not affected by shadow edges, we have introduced the accurate shadow mask by Fredembach and Süsstrunk [2010] in the pairwise potential of our CRF model. By assigning a larger penalty to the shadow edges, we have shown that the object boundaries are detected more accurately.

Overall, this dissertation has shown that introducing NIR information significantly improves the performance of automatic labeling for many classes: Incorporating NIR information outperforms the visible-only strategies for the cases when the key attribute for assigning a certain class to a region is texture (Clothing item) or material (Water), or a combination of texture and material. For the cases where color is the key to recognizing a class, NIR is unlikely to significantly improve the accuracy.

There are different ways to extend and build further on the ideas described in this thesis.

- A database of spectral reflectance of different material classes could delineate the parts of the spectrum where different materials are statistically significantly different. The idea would be to determine where in the spectrum the differences between different material spectra occur and which physical or chemical characteristic contributes to these differences. The reflectance spectra associated with different materials could then be statistically analyzed to determine whether the variance of reflectance between material classes is greater than within classes.

  This information could also be useful if we know the position of the regions in the spectra where the reflectance of each class of material is significantly different from the other classes. This spectral analysis and geometric specification of the most significantly discriminative regions might be applied to design filters in the NIR region; the output of these filters can help reaching a higher classification rate.

- To better understand the advantages of incorporating NIR information and to study the potential of this information, a larger dataset of annotated images with more classes would be beneficial. The semantic segmentation results presented in this dissertation cannot be thoroughly discussed for the classes Soil, Snow or Cell phone as they correspond to the three smallest classes of our dataset. The classifier has only very few

examples to learn the appearance of these categories, and our observation would not be very reliable. Thus, a larger and more balanced dataset would show the advantages of incorporating NIR information more accurately. For another improvement to the dataset, one could acquire a dataset of 4-channel images with only one shot. The border accuracy in the dataset that is used in this dissertation could not be reliably evaluated for some classes due to the movement of the objects between two shots (Sky and Cloud), or the movement of the camera.

- A more thorough study could also be designed to better incorporate shadow information in the semantic image segmentation. From a dataset where shadows as well as the classes are annotated, we could design a model that learns the appearance of the classes both in the illuminated and shadowed parts. The aim would be to incorporate physical properties of the NIR band, such as generally higher reflectances and very marked differences in illuminants' spectral power distribution, to improve the results.

A Energy Minimization with Graph Cuts

Graph cuts can be used to efficiently minimize energies in CRF models. This section is a summary of work in Boykov et al. [2001]. Energies that can be minimized are described in Chapters 3.3 and 7.1 and are of the form

$$\begin{aligned} E(\mathbf{x}) &= E_{un}(\mathbf{x}) + \lambda E_{pair}(\mathbf{x}) \\ &= \sum_{i \in v} \psi_i(x_i) + \lambda \sum_{(i,j) \in \varepsilon} \psi_{i,j}(x_i, x_j) \end{aligned} \tag{A.1}$$

$\psi_i(x_i)$ ensures that the current labeling agrees with the observed data, i.e., it penalizes if the label x_i assigned to the pixel i is not consistent with the observed data. $\psi_{i,j}(x_i, x_j)$ ensures that the current labeling is smooth. Constraints on the pairwise term are explained in Chapter 3.3.

The minimization algorithm that is used in this thesis, α-expansion, is summarized in this section.

A.1 The α-Expansion Minimization Algorithm

In the big picture, the algorithm starts with any initial configuration $\mathbf{x}^0$. $\mathbf{x}^0$ could be set, for instance, by taking for each pixel the label with maximum prior probability $E_{un}(\mathbf{x})$. We then iterate repeatedly over all the possible labels, and try to expand the area covered by the current label (called α). Each pixel i makes a binary decision: it can either keep its old label x_i or switch to the α label, provided that this change decreases the value of the energy function $E(\mathbf{x})$. We repeat this until the labeling does not change anymore.

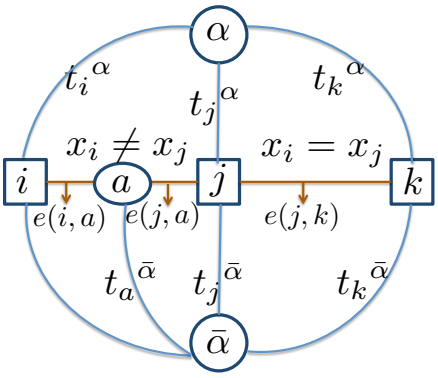


Figure A.1: An α-expansion graph for a 1-dimensional image.

Link	Weight	When
$t_i^{\bar{\alpha}}$	∞	$x_i = \alpha$
	$\psi_i(x_i)$	$x_i \neq \alpha$
t_i^{α}	$\psi_i(x_i)$	always
$e(i,a)$	$\psi_{i,j}(x_i,\alpha)$	$x_i \neq x_j$
$e(j,a)$	$\psi_{i,j}(x_j,\alpha)$	
$t_a^{\bar{a}}$	$\psi_{i,j}(x_i,x_j)$	
$e(j,k)$	$\psi_{j,k}(x_j,x_k)$	$x_j = x_k$

Table A.1: The weights used in the α-expansion algorithm

In more detail, in each iteration, a graph, such as the one in Figure A.1, is constructed with two extra terminal nodes α and $\bar{\alpha}$. Two neighbor pixels i, j with different labels are connected through an intermediate auxiliary node a; neighbor pixels j, k with the same label are directly connected to each other. The nodes are also connected to α and $\bar{\alpha}$. See Table A.1 for all the edge weights. In order to find the optimal α-expansion move, the algorithm seeks the cut with the minimum cost. A cut C is a partitioning of the nodes into two subsets A and $\bar{A}$ such that $\alpha \in A$ and $\bar{\alpha} \in \bar{A}$. The cut's cost is the sum of the weights of the links between A and $\bar{A}$. After finding the minimum-cost cut, a pixel i is assigned the label α if the cut separates i from the terminal α. Otherwise, it keeps its old label.

B Removing Shadows from Images Using Color and NIR

Removing shadows from images can significantly improve and facilitate the performance of certain computer vision tasks, such as tracking, segmentation, and object detection, where shadow boundaries are often confused with those of different surfaces or objects. It is therefore of great importance to discover ways of properly detecting shadows and removing them while keeping other details of the original image intact.

A lot of research has been performed to detect shadows. Finlayson et al. [2002] propose the invariant image method, which requires the knowledge of the capture device's characteristics to color calibrate the camera. Color calibration of the camera leads to 1-dimensional pixel values, as a function of image chromaticities that is invariant under illuminant color and intensity changes. Projecting a color image into the illumination invariant direction forms a gray-scale image, which is independent of the illumination condition. Afterwards, shadow edges are detected by finding edges in the intensity image that are not in the illumination invariant image.

(A) (B) (C)

Figure B.1: Column (A) is the color image. Column (B) shows the NIR image of the scene and column (C) is their corresponding shadow maps.

Levine and Bhattacharyya [2005] propose to manually train a support vector machine to segment an image into shadow and non-shadow regions. After validating the classifier, the shadowed regions are found.

Assuming that the histogram of the illumination is sparse, Weiss [2001] proposes to acquire a sequence of images in which the shadow edges move, i.e., when just illumination changes. The median of this sequence is calculated for each pixel and amounts to the maximum-likelihood estimation of the reflectance only image. Although his method results in very natural looking images, one of the shortcomings of this method is that it is not always practical to acquire a sequence of images where all the objects and surfaces do not move and just changing the illumination generates moving shadow edges.

Fredembach and Süsstrunk [2010] present a simple though accurate shadow detection method (FS method) that employs the inherent sensitivity of digital camera sensors to the near-infrared (NIR) part of the spectrum. Using the property of the NIR band, in which most of the colorants are transparent or have higher reflectance [Burns and Ciurczak, 2001], the authors show that combining the dark map of both visible and NIR images with ratios of the color channels (red, green and blue) to NIR identifies the pixels that are shadow candidates. Figure B.1 shows the shadow map of two images.

As it can be seen in Figure B.1, the results are accurate in real and complex scenes, including regions that are partially lit (penumbra), and not lit at all (umbra).

After identifying shadows in an image, several methods have been proposed to remove these shadows. Levine and Bhattacharyya's method (referred to as "LB" hereafter) is to assign the average color of the lit region to the adjacent shadowed region [Levine and Bhattacharyya, 2005]. Fredembach and Finlayson ("FF") [Fredembach and Finlayson, 2006] form the gradient image and remove the gradient information at the shadow edges. Therefore, the shadow-free image can be obtained by solving a Poisson's equation [Fredembach and Finlayson, 2006, Finlayson et al., 2002].

We hereby propose a new approach to remove the shadows that are detected by applying the FS shadow detection method [Fredembach and Süsstrunk, 2010]: We create a probability map that gives us information on how much shadow each pixel contains. Then, we present a method to lighten up the shadowed regions in order to obtain a shadow free image, which is natural-looking and keeps all the details and textures intact. The results are compared to the LB and FF shadow removal frameworks. This chapter is based on Salamati et al. [2011a].

B.1 Using NIR Information in Detecting Shadows

Prior to removing the shadows, we apply the FS shadow detection framework [Fredembach and Süsstrunk, 2010], that has shown that NIR wavelengths used along with a color image to accurately detect shadows in the image. To this end, we formulate a "dark map" and "ratio map" to find shadow-candidate pixels.

First, a joint dark map of color and NIR images D is formed that is the multiplication of the visible and NIR dark maps (D_{VIS} and D_{NIR}):

$$D_{VIS} = 1 - B; \;\; D_{NIR} = 1 - I_{NIR} \tag{B.1}$$

$$D = f(D_{VIS}) f(D_{NIR}) \tag{B.2}$$

where B is the brightness of the visible image, which is calculated as the pixels' norm in an RGB cube, and I_{NIR} is the intensity of the NIR image. $f(\cdot)$ is an ascending function that compresses the shadows [Fredembach and Süsstrunk, 2010]. The argument is that shadow pixels are to be found in the darker regions of the image. Thus, an "AND" operator on the visible and NIR dark maps identifies the pixels in the image that are dark in the two representations of the scene.

In the second step, the ratio image F is formed to be combined with the dark map and generates a final shadow map M.

$$M = DF \tag{B.3}$$

The physical property of the NIR band (namely, higher reflectance of different materials as well as distinct behavior of most illuminants in the NIR part of the spectrum) set the dark pixels that correspond to dark objects apart from the shadowed pixels in M.

The larger the pixel values in M, the more probable they are to be under the shadow. In [4], it is proposed that the location of the first valley t_1 in the histogram of M specifies the threshold value to generate the final shadow mask S.

$$S(x,y) = \begin{cases} M(x,y) & \text{if } M(x,y) \geq t_1 \\ 0 & \text{if } elsewhere \end{cases} \tag{B.4}$$

We refer the reader to Fredembach and Süsstrunk [2010] for more details about the method.

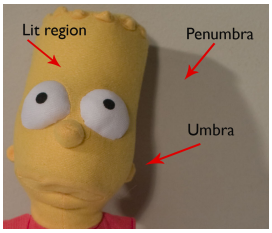


Figure B.2: Umbra and penumbra. A non-point light source will produce three distinct lighting areas; lit regions, partially lit (penumbra), and not lit at all (umbra).

B.2 Shadow Removal Framework

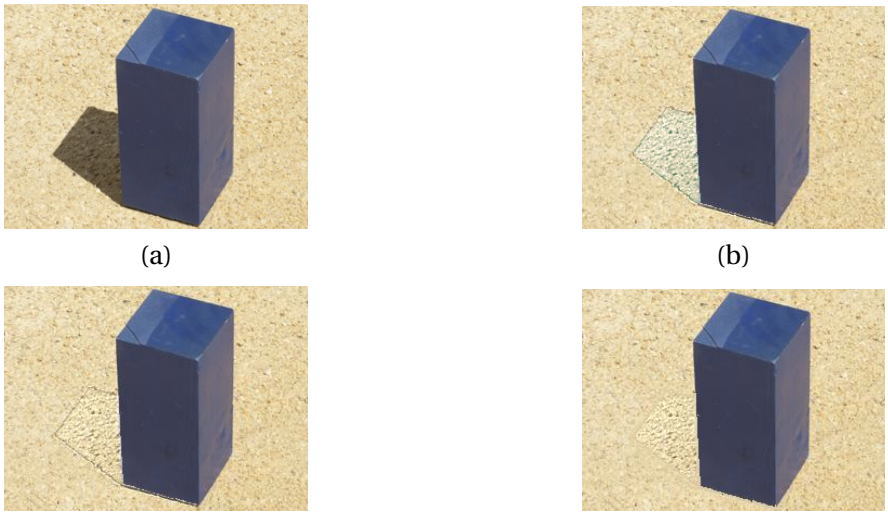
A non-point-light source creates three distinct illuminated areas in a scene, namely non-lit areas (umbra), partially lit areas (penumbra), and completely lit areas. A "shadow probability map" S is formulated in a way that regions that are not lit (a shadowing object blocks all the light from the source) have large values. Pixels in penumbra regions have smaller values in S as the intensity of light is increasing, and the S value in the lit regions is 0. Consequently, S values can be seen as the probability of each pixel to be under the shadow ($S(x \in \text{umbra}) > S(x \in \text{penumbra}) > S(x \in \text{lit regions})$). Figure B.2 shows the different shadow regions in the visible image of a scene. We divide the shadow regions $S(x,y) \neq 0$ into the penumbra and umbra according to their pixel values: Analyzing the histogram of S, we can observe two peaks, which correspond to the umbra and penumbra regions. The position of the valley between these peaks (t_2) gives the threshold to segment S into umbra and penumbra regions.

$$S(x,y) \in \begin{cases} \text{umbra} & \text{if } S(x,y) \geq t_2 \\ \text{penumbra} & \text{if } t_1 \leq S(x,y) < t_2 \\ \text{non-shadow} & \text{if } S(x,y) = 0 \end{cases} \tag{B.5}$$

Removing shadows can be performed by lightening umbra and penumbra regions. Since shadowed regions do not have a constant intensity (the intensity gradually increases from shadow to light), we propose to increase the lightness of the pixels according to the shadow probability map S.

$$L'(x,y) = \begin{cases} L(x,y)g_1(S(x,y)) & \text{if } S(x,y) \in \text{umbra} \\ L(x,y)g_2(S(x,y)) & \text{if } S(x,y) \in \text{penumbra} \\ L(x,y) & \text{if } S(x,y) \in \text{non-shadow} \end{cases} \tag{B.6}$$

where L is the lightness value of I_{VIS} in CIELab color space for the shadowed image, L' is the lightness value of the corrected image, and $g(\cdot)$ is a function used to increase the intensity of



(a) (b) (c) (d)

Figure B.3: (a): Original image. (b): Lightness corrected. (c): Color corrected. (d): Borders corrected.

each pixel under the shadow given the corresponding shadow probability value in S.

$$g_1(s) = se^{s^m}$$
$$g_2(s) = se^{s^n} \tag{B.7}$$

m and n values are parameters that are empirically set at 2 and 4, respectively, for our image dataset.

Figure B.3 illustrates an image where the lightness is corrected applying our method. Increasing the lightness of the shadowed parts using the proposed method keeps the details of the shadowed surfaces intact. As we can see in Figure B.3, although the lightness of the shadowed parts are corrected, some shadowed surfaces in the image still do not look similar to the lit parts. The reason is that, given two pixels with the same surface reflectance, located on opposite sides of a shadow boundary, the ratios of the two pixels are not the same in all three color channels because of the ambient light. These two pixels will differ not only in intensity, but also in hue and saturation. Thus, correcting just the intensity of the shadowed pixels does not remove the shadow and we need to correct the chromaticity values as well. In the following, we refer to a^* and b^* values in CIELAB color space as the chromaticity attributes.

Applying a mean shift algorithm we segment the entire image according to its color values. We start with the penumbra pixels in the color image, having the hypothesis that segments in penumbra are certainly adjacent to a non-shadow region. For each segment in penumbra P we consider all its neighbor segments LIT_j in the lit part of the image ($LIT_j \in$ lit part &

$Di(P) \cap LIT_j \neq \emptyset$, where $Di(\cdot)$ is a dilation function). Among all the neighbor segments, we choose the one that is closest in chromaticity to our segment of interest. We call this segment LIT.

Afterwards, we rescale the shadowed segment's chromaticity values so that the average of the chromaticity in that segment P matches the average of the chromaticity in the aforesaid lit segment LIT.

$$a_P^* := a_P^* \frac{<a_{LIT}^*>}{<a_P^*>}$$
$$b_P^* := b_P^* \frac{<b_{LIT}^*>}{<b_P^*>} \tag{B.8}$$

where a_i^* and b_i^* are the a^* and b^* attributes of the corresponding segment and $<\cdot>$ is the average operator. The chromaticity correction is valid for the surfaces that are partly lit and partly under shadow. For such regions, the chromaticity can be corrected so that the shadow part of the surface will have the same chromaticity as the lit part of the surface. However, there might be surfaces that are completely under shadow. Changing the chromaticity values of them to the closest adjacent lit segment will introduce false colors. To prevent this effect, if the chromaticity difference between P and LIT is not small enough, the chromaticity value of segment P will not be changed.

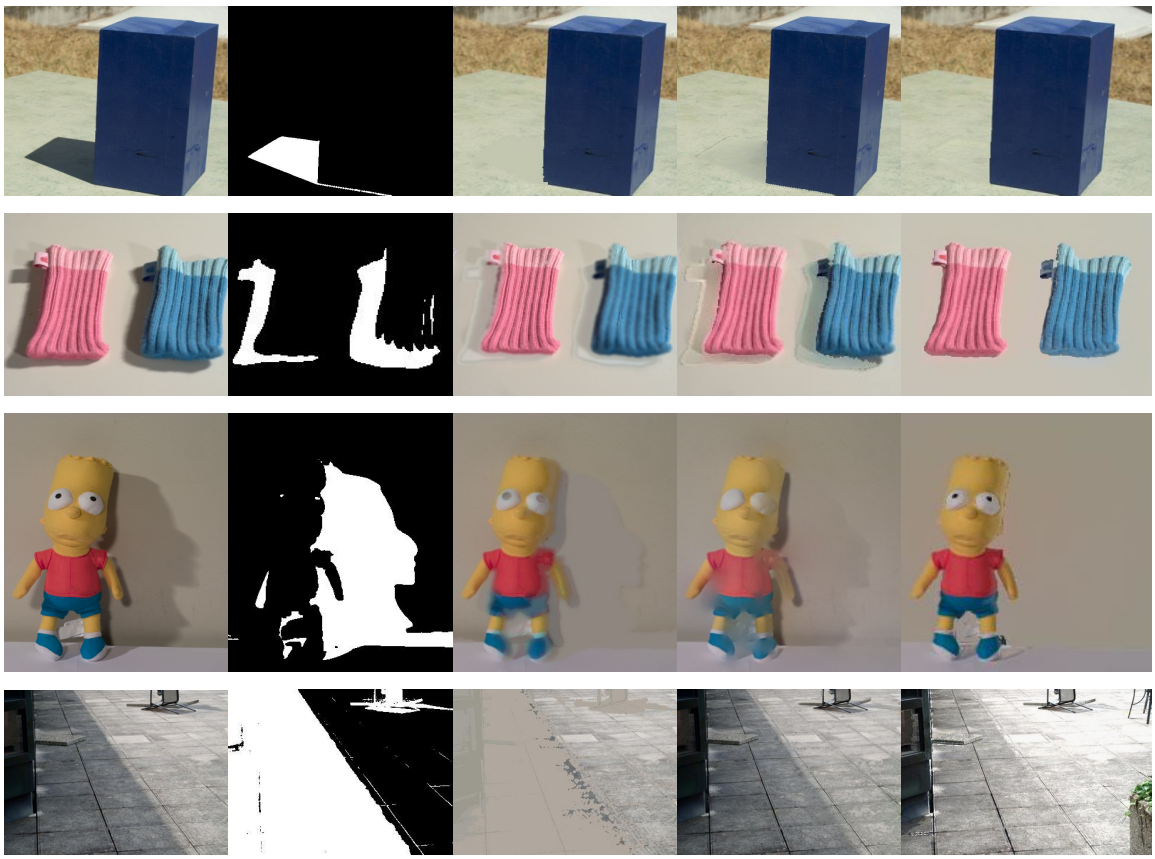
After "correcting" the color of penumbra regions, we continue rescaling the chromaticity values of umbra regions by applying the same technique.

Finally, all boundaries between shadowed regions and neighboring lit regions are smoothed by convolving them with a Gaussian mask. Thus, we introduce a uniform transition between shadowed regions that were lightened and neighboring non-shadowed regions.

B.3 Result and Discussion

Figure B.4 shows the result of applying our framework to some images. The results are compared to the two state of the art shadow removal frameworks in Levine and Bhattacharyya [2005] and Fredembach and Finlayson [2006]. We use the same FS shadow map [Fredembach and Süsstrunk, 2010] for all three algorithms.

We see that the LB framework [Levine and Bhattacharyya, 2005] removes all high frequency details from the shadow regions. The FF results look acceptable but in some cases the overall color is different. This is because of the Neumann boundary conditions in solving the Poisson's



(A) (B) (C) (D) (E)

Figure B.4: Column (A) is the original image, (B) shows the shadow map, (C) shows the results with LB [Levine and Bhattacharyya, 2005], (D) shows the results with FF Fredembach and Finlayson [2006], and (E) shows the results with our algorithm. We can see that the solution we propose preserves not only the colors, but also the textures of the lightened parts.

equation. Additionally, the portion of the information erased at the shadow edges in the gradient image can result in a smudge effect in the re-integrated image. To remove the gradient information on shadow edges, the assumption is that the reflectance on the shadow edges does not change, which is not correct in textured surfaces.

One of the main advantages of our proposed method is that the texture of the surface under the shadow is preserved to a good extent and no harsh transition between the shadowed parts and non-shadowed parts can be seen. The shortcoming of our method is that the result is dependent on the values of the g_1 and g_2 parameters. The proposed values for these parameters do not always give the best result. (See last image in Figure B.4 for a case where the proposed values do not completely remove the shadow.)

B.4 Conclusion

We described a shadow removal method for real images based on the shadow map proposed by Fredembach and Süsstrunk [2010]. We increased the lightness of shadowed regions in an image knowing the shadow probability map. The color of that part of the surface is then corrected so that it matches the lit part of the surface. Our algorithm worked successfully in removing both umbra and penumbra shadows.

Bibliography

R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk. Frequency-tuned salient region detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

J. Bauer, N. Sunderhauf, and P. Protzel. Comparing several implementations of two recently published feature detectors. In *Proceedings of the International Conference on Intelligent and Autonomous Systems*, 2007.

H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *Computer Vision and Image Understanding*, 110:346–359, 2006.

G.A. Blackburn. Hyperspectral remote sensing of plant pigments. *Journal of Experimental Botany*, 58(4):855–867, 2007.

L. Bottou. SGD. http://leon.bottou.org/ projects/sgd/.

Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.

Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

M. H. Brill. Image segmentation by object color: a unifying framework and connection to color constancy. *Journal of the Optical Society of America. A, Optics and image science*, 7(10): 2041–2047, 1990.

M. Brown and S. Süsstrunk. Multispectral SIFT for scene category recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

D. A. Burns and E.W. Ciurczak. *Handbook of Near-infrared analysis*. Marceld Ekkerin,Inc, 2001.

K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *Proceedings of the British Machine Vision Conference*, 2011.

C. M. Christoudias, B. Georgescu, and P. Meer. Synergism in low level vision. In *Proceedings of the International Conference on Pattern Recognition*, 2002.

S. Clinchant, G. Csurka, F. Perronnin, and J.M. Renders. Xrces participation to imageval. In *ImageEval Workshop at CVIR*, 2007.

D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

R. Cook and S. Weisberg. *Residuals and influence in regression*, volume 5. Chapman and Hall New York, 1982.

G. Csurka and F. Perronnin. An efficient approach to semantic segmentation. *International journal of computer vision*, 95(2):198–212, 2011.

G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, page 22, 2004.

M. Everingham, LV Gool, C. Williams, and A. Zisserman. The pascal visual object classes challenge, a. URL http://www.pascalnetwork.org/challenges/VOC.

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL visual object classes challenge 2007 (VOC2007) results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html, b.

L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

L. Fei-Fei, R. VanRullen, C. Koch, and P. Perona. Why does natural scene categorization require little attention? Exploring attentional requirements for natural and synthetic stimuli. *Visual Cognition*, 12(6):893–924, 2005.

G. D. Finlayson and M. S. Drew. 4-sensor camera calibration for image representation invariant to shading, shadows, lighting, and specularities. In *Proceedings of the International Conference on Computer Vision*, 2001.

G. D. Finlayson, M. S. Drew, and B. V. Funt. Color constancy:generalized diagonal transforms suffice. *Journal of the Optical Society of America*, 11(11):3011–3019, 1994.

G. D. Finlayson, S. D. Hordley, and M. S. Drew. Removing shadows from images. *Lecture Notes in Computer Science*, pages 823–836, 2002.

D. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, 2002.

C. Fredembach and G. Finlayson. Simple shadow removal. In *Proceedings of the International Conference on Pattern Recognition*, 2006.

C. Fredembach and S. Süsstrunk. Colouring the near-infrared. In *Proceedings of the IS& T/SID Color Imaging Conference*, 2008.

C. Fredembach and S. Süsstrunk. Automatic and accurate shadow detection from (potentially) a single image using near-infrared information. Technical Report Tech Report 165527, EPFL, 2010.

B. V. Funt and G. D. Finlayson. Color constant color indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):522–529, 1995.

M. S. Drew G.D. Finlayson, S.D. Hordley and C. Lu. On the removal of shadows from images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:59–68, April 2006.

J. Geusebroek, R. van den Boomgaard, A. Smeulders, and H. Geerts. Color invariance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(12), December 2001.

P. Gunawardane, T. Malzbender, R. Samadani, A. McReynolds, D. Gelb, and J. Davis. Invisible light: Using infrared for video conference relighting. In *Proceedings of IEEE International Conference on Image Processing*, 2010.

Z. Guoying, H. Xiaohua, T. Matti, Z. L. Stan, and P. Matti. Facial expression recognition from near-infrared videos. *Image and Vision Computing*, 29(9):607 – 619, 2011.

G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *Proceedings of the European Conference on Computer Vision*, 2008.

A. Criminisi J. Winn and T. Minka. Object categorization by learned universal visual dictionary. In *Proceedings of the International Conference on Computer Vision*, 2005.

T. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *Proceedings of the Conference on Neural Information Processing Systems*, 1999.

F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *Proceedings of the International Conference on Computer Vision*, 2005.

P. Kohli and M.P. Kumar. Energy minimization for linear envelope mrfs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

P. Kohli, L. Ladickỳ, and P.H.S. Torr. Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, 82(3):302–324, 2009.

V. Kolmogorov and R. Zabin. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004.

S.G. Kong, J. Heo, B.R. Abidi, J. Paik, and M.A. Abidi. Recent advances in visual and infrared face recognition a review. *Computer Vision and Image Understanding*, 97, 2005.

S. M. Kosslyn. Information representation in visual images. *Cognitive Psychology*, 7, 1975.

P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Proceedings of the Conference on Neural Information Processing Systems*, 2011.

D. Krishnan and F. Fergus. Dark flash photography. In *ACM Transactions on Graphics, SIGGRAPH 2009 Conference Proceedings*, 2009.

B. Krishnapuram, L. Carin, M.A.T. Figueiredo, and A.J. Hartemink. Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(6):957–968, 2005.

A. Kulcke, C. Gurschler, G. Spock, R. Leitner, and M. Kraft. On-line classification of synthetic polymers using near infrared spectral imaging. *Journal of Near-infrared Spectroscopy*, 11(1): 71–81, 2003.

L. Ladicky, C. Russell, P. Kohli, P. H. S. Torr, and O. Brookes. Graph cut based inference with co-occurrence statistics. In *Proceedings of the European Conference on Computer Vision*, 2010.

S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

M. D. Levine and J. Bhattacharyya. Removing shadows. *Pattern Recognition Letters*, 26(3): 251–265, 2005.

S. Z. Li, R. Chu, M. Ao, L. Zhang, and R. He. Highly accurate and fast face recognition using near infrared images. In *Advances in Biometrics*, pages 151–158. 2005.

L. Li-Jia, R. Socher, and L. Fei-Fei. Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

D. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the International Conference on Computer Vision*, 1999.

D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

J. Mairal, M. Leordeanu, F. Bach, M. Hebert, and J. Ponce. Discriminative sparse image models for class-specific edge detection and image interpretation. In *Proceedings of the European Conference on Computer Vision*, 2008.

K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60:63–86, 2004.

A. Oliva and A. Torralba. http://people.csail.mit.edu/torralba/code/spatialenvelope/.

A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42:145–175, 2001.

F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007.

F. Perronnin, J. Sánchez, and Thomas Mensink. Improving the Fisher Kernel for large-scale image classification. In *Proceedings of the European Conference on Computer Vision*, 2010.

N. Plath, M. Toussaint, and S. Nakajima. Multi-class image segmentation using conditional random fields and global classification. In *Proceedings of the Annual International Conference on Machine Learning*, 2009.

J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Proceedings of Advances in Large Margin Classifiers*, 1999.

C. Pohl and J.L. Van G. Review article multisensor image fusion in remote sensing: concepts, methods and applications. *International Journal of Remote Sensing*, 19(5):823–854, 1998.

C. Qiang, S. Zheng, L. Si, C. Xiangyu, Y. Xiaotong, C. Tat-Seng., Y. Shuicheng, H. Yang, H. Zhongyang, and S. Shengmei. Boosting classification with exclusive context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognitio*, pages 1585–1592, 2011.

T. Randen and J. H. Husoy. Filtering for texture classification: A comparative study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4):291–310, 1999.

C. Rother, V. Kolmogorov, and A. Blake. "GrabCut": interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 2004.

D. M. Rouse and S. S. Hemami. Quantifying the use of structure in cognitive tasks. In *Proceedings of IS&T/SPIE Electronic Imaging: Human Vision and Electronic Imaging*, 2007.

Z. Sadeghipoor, Y.M. Lu, and S. Süsstrunk. Correlation-based joint acquisition and demosaicing of visible and near-infrared images. In *Proceedings of IEEE International Conference on Image Processing*, 2011.

N. Salamati and S. Süsstrunk. Material-based object segmentation using near-infrared information. In *Proceedings of the IS& T/SID Color Imaging Conference*, 2010.

N. Salamati, C. Fredembach, and S. Süsstrunk. Material classification using color and NIR images. In *Proceedings of the IS& T/SID Color Imaging Conference*, 2009.

N. Salamati, A. Germain, and S. Süsstrunk. Removing shadows from images using color and near-infrared. In *Proceedings of IEEE International Conference on Image Processing*, 2011a.

N. Salamati, D. Larlus, and G. Csurka. Combining visible and near infrared cues for image categorisation. In *Proceedings of the British Machine Vision Conference*, 2011b.

N. Salamati, D. Larlus, G. Csurka, and S. Süsstrunk. Semantic image segmentation using visible and near-infrared channels. In *Workshop on Color and Photometry in Computer Vision, ECCV*, 2012.

J. Salvi, J. Pages, and J. Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827–849, 2004.

M. Sassi, K. Vancleef, B. Machilsen, S. Panis, and J. Wagemans. Identification of everyday objects on the basis of gaborized outline versions. In *Proceedings of European Conference on Visual Perception*, 2010.

L. Schaul, C. Fredembach, and S. Süsstrunk. Color image dehazing using the near-infrared. In *Proceedings of IEEE International Conference on Image Processing*, 2009.

H. R. Sheikh, A. C. Bovik, and L. Cormack. No-reference quality assessment using natural scene statistics: Jpeg2000. *IEEE Transactions on Image Processing*, 14(11):1918–1927, 2005.

L. Shen, J. He, S. Wu, and S. Zheng. Face recognition from visible and near-infrared images using boosted directional binary code. In *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*, pages 404–411. 2012.

J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *Proceedings of the European Conference on Computer Vision*, 2006.

J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, 2003.

G. W. Snedecor and W. G. Cochran. *Statistical Methods*. Iowa State University Press, 1989.

H. Sun, C. Wang, B. Wang, and N. El-Sheimy. Unsupervised video-based lane detection using location-enhanced topic models. *Optical Engineering*, 49(10):107201–107201, 2010.

S. Süsstrunk, C. Fredembach, and D. Tamburrino. Automatic skin enhancement with visible and near-infrared image fusion. In *Proceedings of the International Conference ACM Multimedia*, 2010.

R. Szeliski. Image alignment and stitching: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 2(1):1–104, 2006.

D. S. Taubman, M. W. Marcellin, and M. Rabbani. Jpeg2000: Image compression fundamentals, standards and practice. *Journal of Electronic Imaging*, 11(2):286–287, 2002.

A. Torralba. Contextual priming for object detection. *International Journal of Computer Vision*, 53(2):169–191, 2003.

S. Ullman and G. J. Power. High-level vision: Object-recognition and visual cognition. *Optical Engineering*, 36(11):3224–3224, 1997.

K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32 (9):1582–1596, 2010.

J. C. van Gemert, C. J. Snoek, C. G.M .and Veenman, A. W.M. Smeulders, and J.M. Geusebroek. Comparing compact codebooks for visual categorization. *Computer Vision and Image Understanding*, 114(4):450–462, 2010.

V. Vapnik. *The Nature of Statistical Learning Theory*. springer, 1999.

J. Verbeek and B. Triggs. Region classification with markov field aspects models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

J. Vogel and B. Schiele. Semantic modeling of natural scenes for content-based image retrieval. *International Journal of Computer Vision*, 72:133–157, April 2007.

V. Walter. Object-based classification of remote sensing data for change detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 58:255–238, 2004.

J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

Y. Weiss. Deriving intrinsic images from image sequences. In *Proceedings of the International Conference on Computer Vision*, 2001.

J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

X. Zhang, T. Sim, and X. Miao. Enhancing photographs with near infrared images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

W. Zhou, G. Huang, Au. Troy, and M. L. Cadenasso. Object-based land cover classification of shaded areas in high spatial resolution imagery of urban areas: A comparison study. *Remote Sensing of Environment*, 113(8):1769–1777, 2009.

X. Zhou, K. Yu, T. Zhang, and T. Huang. Image classification using super-vector coding of local image descriptors. In *Proceedings of the European Conference on Computer Vision*, 2010.

Neda Salamati

CONTACT INFORMATION

BC 318- Image and Visual Representation Group (IVRG)
School of Computer and Communication Sciences (IC)
Ecole Polytechnique Fédérale de Lausanne (EPFL)
Station 14, Lausanne 1015, Switzerland

Tel: (41 21) 6937-604
Mobile: (41 78) 8191-280
E-mail: neda.salamati@epfl.ch
WWW: http://ivrg.epfl.ch/people/salamati

RESEARCH INTERESTS

Semantic image segmentation

Image categorization

Color science

Machine learning applied to computer vision

EXPERIENCE

Xerox Research Centre Europe(XRCE), Grenoble, France
Research assistant intern **March-September 2011**

Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
Teaching assistant, Courses: Digital photography, Color imaging, Foundation of imaging science
September 2009-July 2012

EDUCATION

Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland

Ph.D. Candidate, Computer science, since Oct. 2008

- Advisor: Prof. Sabine Süsstrunk
- Expected graduation date: May 31, 2013

Tehran Polytechnic, Tehran, Iran

M.S., Textile eng., GPA: 17.07/20, Sept. 2005 - Sept. 2007

Tehran Polytechnic, Tehran, Iran

B.S., Textile eng., GPA: 17.69/20, Sept. 2001 - Sept. 2005

HONORS AND AWARDS

Silver Award for best Teaching Assistants in IC/EPFL for the academic year 2011/2012

Best Intern Award for my presentation on "Combining Visible and Near-infrared Cues for image Categorization", June 2011 at Xerox Research Centre Europe

Cactus Award for Best Interactive Paper at IS&T/SID 18th Color Imaging Conference (CIC), 2010

Second rank during my master studies

PUBLICATIONS

N. Salamati, D. Larlus, G. Csurka and S. Süsstrunk, Semantic Image Segmentation Using Visible and Near-Infrared Channels, Proc. European Conference on Computer Vision (ECCV), 2012.

N. Salamati, Z. Sadeghipoor and S. Süsstrunk, Compression of Multispectral Images: Color (RGB) plus Near-Infrared (NIR), Proc. IEEE International Workshop on Multimedia Signal Processing (MMSP), 2012.

N. Salamati, A. Germain and S. Süsstrunk, Removing Shadows from Images Using Color and Near-infrared, Proc. IEEE International Conference on Image Processing (ICIP), 2011.

N. Salamati, D. Larlus and G. Csurka, Combining Visible and Near Infrared Cues for image Categorisation, Proc. British Machine Vision Conference (BMVC), 2011.

N. Salamati, Z. Sadeghipoor Kermani and S. Süsstrunk, Analyzing Near-infrared Images for Utility Assessment, Proc. IS&T/SPIE Electronic Imaging: Human Vision and Electronic Imaging, 2011.

C. Fredembach, N. Barbuscia, Y. Lu, N. Salamati, L. Schaul, D. Tamburrino and S. Süsstrunk, Enhancing the Visible with the Invisble: RGB plus Near-Infrared, Presented at: IEEE International Conference on Computational Photography (ICCP), 2010.

N. Salamati and S. Süsstrunk, Material-Based Object Segmentation Using Near-Infrared Information, Proc. IS&T/SID 18th Color Imaging Conference (CIC), 2010.

N. Salamati, C. Fredembach and S. Süsstrunk, Material Classification Using Color and NIR Images, Proc. IS&T/SID 17th Color Imaging Conference (CIC), 2009.

N. Salamati and A. H. Amirshahi, The Comparison between PCA and Simplex Methods for Reflectance Recovery, Proc. AIC07 midterm Meeting, 2007.

Paper in preparation

N. Salamati, D. Larlus, G. Csurka and S. Süsstrunk, A Study on Incorporating NIR Information into a Better Semantic Segmentation.

Patent

N. Salamati, D. Larlus, G. Csurka and C. Saunders, Graph-Based Object Segmentation Integrating Visible and Near-Infrared Information, US patent pending.

Skills

Programming languages: C++ (general knowledge), Matlab.

Natural Languages: English, French (good knowledge), Persian (mother tongue).