Specular Surface Reconstruction from Images

Thomas Bonfort

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

N° attribué par la bibliothèque
/_/_/_/_/_/_/_/_/_/_/

THESE

pour obtenir le grade de
DOCTEUR DE L'INPG

Spécialité : Imagerie Vision et Robotique
préparée au laboratoire **GRAVIR**
dans le cadre de l'Ecole Doctorale :
Mathématiques, Sciences et Technologies de l'Information, Informatique

———

présentée et soutenue publiquement par

Thomas BONFORT

le 20 février 2006

RECONSTRUCTION DE SURFACES RÉFLÉCHISSANTES À PARTIR D'IMAGES

———

directeurs de thèse : Peter Sturm et Radu Horaud

———

JURY

Président :	**Roger Mohr**
Rapporteurs :	**Tomáš Pajdla**
	Francis Schmitt
Examinateurs :	**Ryad Benosman**
	Peter Sturm

en souvenir de Manu,

Remerciements

Un grand merci tout d'abord à Peter Sturm pour m'avoir encadré au cours de cette thèse, et dont la disponibilité et la compétence ont été capitales à son bon déroulement. Je suis particulièrement reconnaissant pour la confiance et la grande liberté qu'il m'a accordé au cours de ces années passées à travailler ensemble.

Je souhaite remercier Roger Mohr, pour avoir accepté de présider mon jury de thèse, ainsi que Radu Horaud pour m'avoir accueilli dans de si bonnes conditions dans son équipe. Je remercie Tomáš Pajdla et Francis Schmitt, rapporteurs, ainsi que Ryad Benosman, examinateur, pour avoir porté de l'intérêt à ces travaux et pour leurs remarques constructives et encourageantes.

Enfin, je salue amicalement les thésards et permanents de l'équipe avec qui j'ai partagé des moments plus ou moins studieux, en particulier et par ordre totalement arbitraire Aude, JS, Pau, Jumo et Sri.

Abstract

Specular surface reconstruction from images is a relatively little explored topic, due to the rareness of such objects and the inherent complexity in the induced geometry compared to that of matte shading. This is due to the fact that the apparent texture of the surface is dependant on the viewpoint, or in other words, that the light path from a feature point to a camera pixel is not a straight line. As such, these surfaces are ignored or considered as noise in most 3D reconstruction techniques, whereas we show that the constraints they can provide can be used to compute accurate geometric information on position and orientation. This thesis presents two methods for obtaining position and orientation of points of a specular surface from the reflection of known environning points, without making the usual smoothness or continuity assumptions of nearly all prior methods. The first of these methods is an extension to the space carving framework, which computes voxels belonging to a specular surface by relying on geometric consistency measures rather than photometric ones, which in this case do not make sense. The second method is based on triangulation, and requires a fixed camera viewing the reflection of at least 2 environning points for each surface point to be reconstructed. Finally, we present methods for estimation of the pose of calibration objects when they are not in direct view of a camera, through the reflection by specular objects. The first of these methods assumes that the calibration object is seen through 3 or more mirrors known to be planar, while the second presents a theoretical constraint for the obtention of the pose of such an object placed at two locations, when seen through the reflection of an arbitrary surface.

Keywords: Specular surfaces, Reconstruction, Pose estimation

Résumé

La reconstruction de surfaces spéculaires à partir d'images est un domaine relativement peu exploré, du fait du caractère peu commun de ces objets, et la complexité induite par rapport aux surfaces mattes. Ceci est du au fait que la texture apparente de telles surfaces est dépendante du point de vue, ou formulé autrement, que le chemin lumineux entre un point d'intérêt et un pixel n'est une ligne droite. De ce fait, la plupart des algorithmes de reconstructions ignorent les contributions spéculaires, alors que nous montrons que le contraintes qu'elles apportent permettent d'obtenir des informations géométriques de localisation et d'orientation précises, et ce sans les contraintes de continuité ou de régularité habituellement requises. Cette thèse présente deux méthodes permettant d'obtenir la position et l'orientation de points d'une surface parfaitement spéculaire, à partir de la réflection de points environnants connus. La premiére étend les approches de "space carving", et obtient des voxels d'un objet spéculaire en utilisant une mesure de consistance géométrique putôt que photométrique, qui dans ce cas n'a pas de sens. La deuxième procède par triangulation, en supposant une caméra fixe observant la refléction d'au moins 2 points connus par point de la surface à reconstruire. Finalement, nous proposons des méthodes pour obtenir la pose d'objets de calibration alors qu'ils ne sont pas dans le champ de vue d'une caméra, à travers la reflection d'objets spéculaires. La première suppose que cet objet est vu à travers la réflection de 3 miroirs plans inconnus, et obtient par ailleurs la pose de ces miroirs. La seconde présente une contrainte géométrique permettant théoriquement d'obtenir la pose d'un tel objet placé à deux endroits différents, vu à travers la reflection d'une surface spéculaire quelconque.

Mots Clés: Surfaces spéculaires, Reconstruction, Estimation de pose

Résumé en Français

Contexte de la vision par ordinateur

La plupart des efforts de la communauté de vision par ordinateur tendent à l'obtention de modèles géométriques d'objets à partir d'images. Bien que le système visuel humain excelle dans ce domaine, reconnaissant immédiatement de multiples objets, obtenant la structure tridimensionnelle de n'importe quel environnement, ou estimant la réflectance de la plupart des types de surfaces, la mise au point d'un système artificiel similaire reste un domaine de recherche actif et ouvert. Les modèles numériques en eux même possèdent de nombreux avantages par rapport aux objets concrets qu'ils représentent : ils peuvent être inspectés à toute résolution, affichés dans des environnements variés, stockés et indexés dans des bases de données, transmis télématiquement, modifiés dans des applications de CAO, etc... Les domaines d'application pour ces modèles incluent l'archéologie, la chirurgie, l'aide aux handicapés, la robotique, le divertissement ou le contrôle qualité. L'obtention de tels modèles numériques est dès à présent possible, mais requiert à la fois du temps d'acquisition et un matériel onéreux et peu adapté à une utilisation de masse : un palpeur placé sur un bras robotique permet l'acquisition de points 3D de manière très précise, avec l'inconvénient majeur de devoir physiquement interagir avec la surface à mesurer, et de même, un goniomètre permet l'acquisition de la réflectance complète d'un type de surface donné. L'utilisation d'un scanner laser permet d'accélérer le processus d'acquisition et d'éviter le contact avec la surface à reconstruire, mais demeure un appareillage complexe et cher, et ne permet pas, notamment pour le cadre de cette thèse, la mesure de surfaces hautement spéculaires.

L'obtention de tels modèles en utilisant uniquement des images est une perspective attrayante, de part le fait que les appareils produisant ces images à partir d'observations sont à la fois courants, peu coûteux et non invasifs. De plus, le problème semble résoluble étant donné que c'est une tâche courante du système visuel humain, qui intègre à chaque instant des informations physiologiques (parallaxe, convergence, parallaxe de mouvement) et psychologiques (échelle, perspectives, gradients de texture, ombrages...) afin d'obtenir une représentation précise de son environnement. La complexité inhérente à toutes ces informations est trop importante pour être à la fois interprétée et traitée

par un système artificiel, d'où le fait que l'état de l'art de la vision par ordinateur se limite à des sous classes du problème dans sa globalité. Par exemple, les algorithmes de reconstruction à partir d'image sont généralement limités aux surfaces parfaitement mattes, les algorithmes d'estimation de réflectance supposent une géométrie connue et un nombre restreint de réflectances possibles, et les algorithmes de reconnaissance se limitent à un petit nombre de classes d'objets reconnaissables.

Dans le sous domaine dans lequel se situe cette thèse, l'obtention de modèles géométriques à partir d'images, le but est d'obtenir une représentation numérique de l'occupation de l'espace par un objet, ou en d'autres mots les coordonnées en 3 dimensions des points qui le constituent. La plupart des méthodes de l'état de l'art font une hypothèse draconienne sur la réflectance de la surface constituant l'objet, en supposant que l'intensité lumineuse observée à un point donné est indépendante du lieu d'observation : cette propriété est vérifiée en partie par le bois brut, le papier, les plâtres et bétons, et certains tissus. Pour les méthodes utilisant la mise en correspondance, comme la stéréo, le structure from motion, ou à moindre mesure le space carving, cette hypothèse sert à permettre l'identification et l'appariement de points correspondants au travers les différentes images de l'acquisition. L'informaton géométrique apportée par cet appariement permet l'obtention de contraintes sur la structure de la surface correspondante. Dans le cadre des algorithmes de shape from shading, cette contrainte permet de dériver l'orientation d'une surface étant donné son intensité et la position de la source lumineuse.

Présentation du problème

La plupart des surfaces qui nous entourent, en revanche, ne vérifient pas cette hypothèse. Les techniques qui supposent des surfaces mattes produisent des résultats aberrants si leurs données ne vérifient pas ce modèle, ce qui est le cas avec les surfaces réfléchissantes :

- les méthodes procédant par appariement reconstruisent en général des points derrière ou devant la surface véritable selon si la surface est convexe ou concave.
- les méthodes de shape from silhouette peinent lors de l'extraction des silhouettes aux angles rasants.
- l'énergie transportée par le rayon d'un scanner laser est reflétée dans la direction miroir et n'atteint pas le capteur associé.

De ce fait, les spécularités sont soit traitées comme du bruit, produisant des résultats erronés, soit éliminées dans une étape de pré-traitement, soit évitées lors de l'acquisition en imposant des contraintes fortes sur les positions des objets, sources de lumière et caméras.

Pour autant, les techniques ne permettant de traiter que la réflection spéculaire sont tout autant limitantes dans leur champ d'application. Les objets exhibant une composante spéculaire nette sont rares, et donc des méthodes s'y limitant peuvent paraître surprenantes. Il existe quand même une classe d'applications où un tel problème apparaît directement (voir illustrations page 37) :

- Essentiellement pour le contrôle qualité de pièces usinées, notamment dans l'industrie automobile pour les tôles et les éléments vitrés, où la flexibilité et la rapidité de mesures à partir d'images pourraient permettre un contrôle en temps réel sur les chaînes de production.
- Pour la création et le contrôle des pièces optiques pour télescopes et appareils photos.
- Plus anécdotiquement pour la réalité virtuelle ou la digitalisation d'objets culturels.

Le cœur du problème est bien évidemment ailleurs, et les travaux sur la reconstruction de surfaces réfléchissantes doivent plutôt être vus comme une étape vers des méthodes de reconstruction génériques, permettant de traiter à la fois des surfaces mattes, spéculaires et hybrides.

Géométrie inhérente à la réflection spéculaire

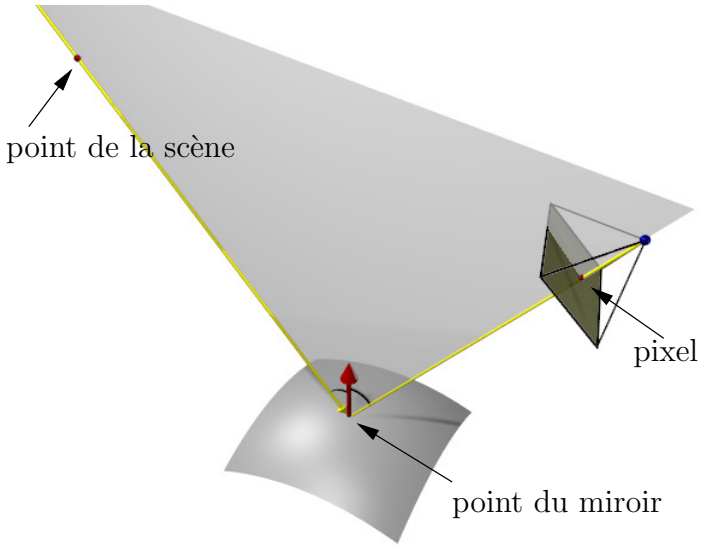
Toute la difficulté inhérente aux surfaces réfléchissantes provient du fait que le chemin emprunté par la lumière lors d'une réflection pour un pixel donné est composé de deux segments de droite, illustrés sur la figure page 12, et définis par les points de l'espace que l'on dénommera :

- **point du miroir** qui est le point en 3 dimensions appartenant à la surface, et où a lieu la réflection. Ce point n'a pas de texture propre, son apparence dépend du point d'observation et de l'environnement.
- **pixel** qui est un point du plan image de la caméra, et la projection du point du miroir.
- **point de la scène** qui est le point dont la "texture" est imagée sur le pixel.

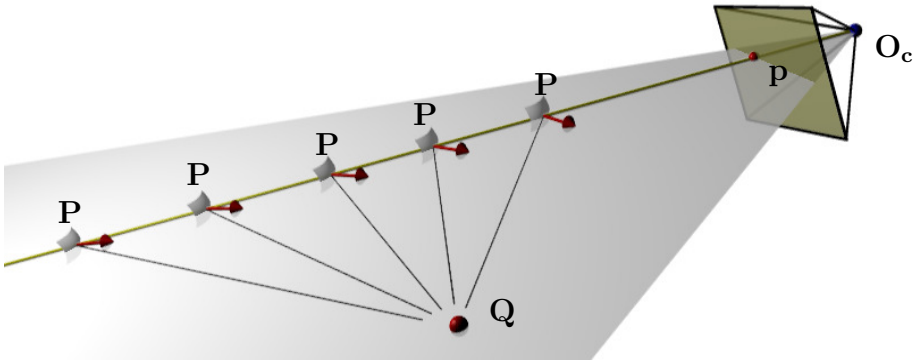
Dans le cas déjà très contraint où une caméra calibrée observe la réflection par une surface spéculaire d'un point de la scène de coordonnées connues, il y a une famille unidimensionnelle de positions et d'orientations de surfaces le long du rayon de projection qui correspondent à cette observation, comme illustré sur la figure page 12.

État de l'art

Les surfaces spéculaires ont été un centre d'intérêt de plusieurs travaux de la communauté vision par ordinateur depuis le début des années 80, dans la plupart des cas dans un but de détection et d'élimination pour des algorithmes traitant les surfaces mattes.



Chemin emprunté par la lumière formé de deux segments de droite reliant le point de la scène, le point du miroir et le pixel de la caméra.



Ambiguïté sur la reconstruction Étant donnée la correspondance entre un pixel $\mathbf{p}$ et un point de la scène $\mathbf{Q}$, il existe une famille unidimensionnelle de surfaces $\{\mathbf{P}\}$ le long du rayon de projection vérifiant la contrainte de réflection.

Dans le domaine central à cette thèse, les méthodes de reconstruction peuvent être catégorisées en deux grandes familles, la première utilisant des contraintes géométriques pour directement calculer des informations sur la surface, et la deuxième procédant par intégration autour de points initiaux.

Les approches géométriques Oren and Nayar [ON95b] utilisent le suivi d'un point de la scène pour retrouver le profile d'une surface lisse, dans le cas où le centre optique de la caméra, le point de la scène et le point du miroir où a lieu la réflection sont coplanaires. Le résultat ne s'étend pas au cas 3-dimensionnel sans a priori fort sur la surface. Zheng *et al.* [ZM98] observent les reflets produits par deux sources circulaires dans un objet en rotation sur une table. Les résultats présentés manquent malheureusement de précision pour être exploitables. Dans un contexte stéréoscopique, Knauer *et al.* [KLKH05] utilisent une contrainte similaire à celle du chapitre 2 pour entreprendre à chaque une pixel une optimisation sur la profondeur du point du miroir correspondant. Savarese *et al.* [SP02],[SCP04] détaillent les contraintes mathématiques lors de la réflection de plusieurs droites concourantes, en utilisant des mesures de courbure et d'échelle, ce qui limite le nombre de points qu'il est possible de reconstruire.

Les méthodes présentées dans cette thèse aux chapitres 2 et 3 font partie de cette classe d'approches.

Les approches par intégration Lorsque des correspondances relativement denses entre pixels et points de la scène peuvent être établies, une méthode souvent utilisée est d'intégrer autour de points du miroir initiaux connus, en utilisant soit des modèles paramétriques de surfaces, soit uniquement des points et normales. Ces méthodes ne requièrent qu'une correspondance par point à reconstruire, mais ont tendance à accumuler les erreurs en s'éloignant des points initiaux. Une implémentation caractéristique est donnée par Schultz dans [Sch96] appliquée à la reconstruction de la surface de l'océan. Dans [HBKM96], Halstead *et al.* présentent un système permettant la reconstruction de la cornée humaine à partir d'images, en ajustant une surface spline. Dans ce scénario très contraint la précision obtenue est inférieur au micron. Similairement, Tarini *et al.* [TLGS03] obtiennent une reconstruction de surface lisse sans explicitement recourir à une initialisation, en minimisant une erreur d'autocohérence sur la surface.

Perception par l'humain La perception et l'interprétation par le système visuel humain au travers uniquement la réflection spéculaire (c'est a dire sans contours occultants ou ombrage) est un sujet ouvert. Blake *et al.* [BB90] montre que le système visuel humain peut interpréter les informations de profondeur apportées par les spécularités. Dans [SFFP04], Savarese *et al.* concluent que le système visuel humain n'est pas meilleur que le hasard dans le cas de l'observation de la réflection de grilles régulières. Flemming *et*

al. in [FTA04] arrivent à la conclusion opposée lorsque les scènes réfléchies sont des environnements naturels réels. Norman *et al.* [NTO04] remarquent aussi que le cadre le plus favorable à la reconnaissance est lors de la présence de spécularités.

Métrologie L'inspection de surfaces est une tâche courante dans les environnements industriels, en particulier dans l'industrie automobile dans le cadre des surfaces réfléchissantes pour les tôles et les parebrises. Traditionnellement cette tâche est effectuée soit par palpeur, soit par scanner laser après avoir rendu la surface non réfléchissante (par peinture ou poussière de craie). Des techniques permettant d'obtenir des résultats similaires avec des techniques uniquement image, et respectant des contraintes de temps d'acquisition suffisamment faibles, trouveraient probablement des applications sur ces chaînes de production.

Récemment un scanner laser spécifique a été développé par [BOIK01], et consiste en une tête mobile et un système de plaques permettant de ne capter l'énergie du laser que dans des directions spécifiques, avec l'inconvénient majeur de temps d'acquisition élevés. Par ailleurs, dans le cadre de l'inspection de finition (courbures locales uniquement), les systèmes à base de déflectometrie [KKH04] donnent des résultats probants, mais sans aller jusqu'à un modèle tridimensionnel de la surface.

Contributions de la thèse

Les contributions et le plan de cette thèse sont les suivants :

- Le chapitre 2 présente une méthode de reconstruction s'incorporant dans le cadre des méthodes de space carving, en présentant une mesure de consistance géométrique plutôt que photométrique.
- Le chapitre 3 présente un contrainte géométrique permettant la reconstruction de surfaces réfléchissantes quelconques par simple intersection, en utilisant une caméra fixe et des correspondances provenant de deux ou plus positions différentes.

Ces deux méthodes permettent la reconstruction de surfaces spéculaires quelconques, sans avoir à faire l'hypothèse habituelle de continuité ou de dérivabilité. La précision obtenue par la deuxième méthode semble être suffisante pour de nombreuses applications.

Un problème récurrent dans le cadre de la reconstruction de surfaces réfléchissantes est l'obtention de correspondances entre pixels de l'image et points de la scène, que l'on considère calibrés, c'est à dire ayant des coordonnées connues. L'obtention de la pose de ces points de la scène est un problème pratique non trivial, étant donné qu'ils ne sont en général pas dans le champ de vue de la caméra correspondante :

- Le chapitre 4 présente des méthodes d'estimation de pose d'un objet de calibrage, lorsque cet objet n'est pas dans le champ de vue de la caméra en question.

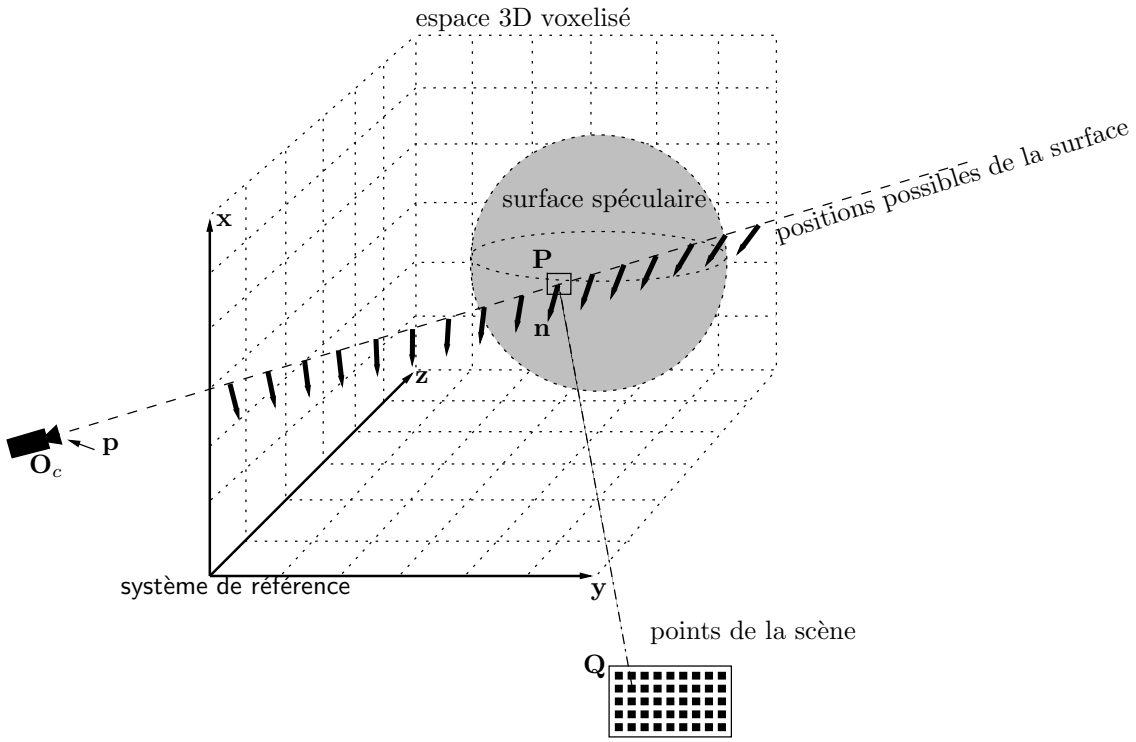
Approche volumétrique

Ce chapitre propose une extension aux approches de reconstruction dites volumiques ou par space carving, initialement présentées par Seitz et Dyer [SD97] et Kutulakos et Seitz [KS00]. Plutôt que d'utiliser des contraintes obtenues par l'appariement au travers plusieurs images de points à reconstruire, ces méthodes utilisent une discrétisation de l'espace à reconstruire, et pour chacun des éléments (appelés voxels) de cette discrétisation, vérifient si leurs observations (*i.e.* leurs projections dans les images) sont cohérentes entre elles. Si ces observations sont cohérentes, l'élément est considéré comme appartenant à la surface à reconstruire, sinon il est rejeté. L'avantage majeur de ces méthodes est qu'elles ne nécessitent pas de mise en correspondance explicite entre les points image correspondant aux points de l'objet à reconstruire, mais au dépens de la précision des résultats obtenus. Pour les surfaces mattes, la cohérence d'un élément est quantifiée par la disparité des couleurs qui lui sont associées, une mesure inapplicable pour les surfaces spéculaires, où la couleur observée dépend de l'environnement et du point de vue. A la place de cette invariance photométrique, nous proposons d'utiliser une invariance géométrique, à savoir que l'orientation d'une surface est indépendante du point de vue.

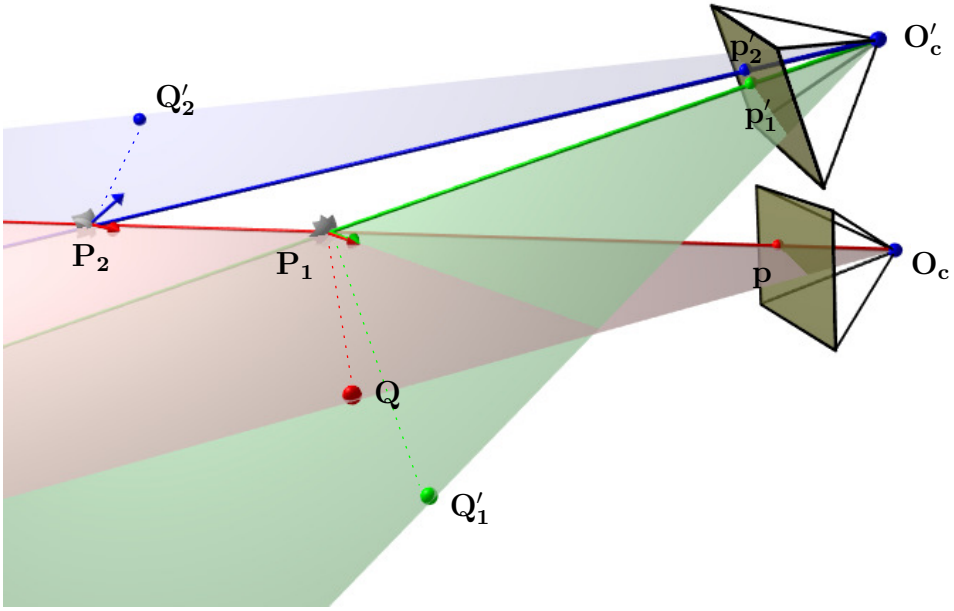
L'information donnée pour une correspondance et un point de vue de caméra est illustrée dans la figure page 16. Pour chacun des voxels traversés par le rayon de projection correspondant, la loi de la réflection permet d'associer la normale à une surface passant par ce voxel et produisant la même observation. Lorsque l'on dispose de plusieurs vues et de plusieurs correspondances, il est donc possible de différencier un voxel intersectant la surface spéculaire d'un voxel "vide" en utilisant une mesure de consistance sur les normales associées aux voxels de l'espace, comme illustré sur la figure page 17.

Mesures de consistance géométrique

Avec des données exactes et une discrétisation de l'espace infiniment précise, un voxel peut être rejeté ou conservé en vérifiant si ses normales associées sont égales entre elles ou non. En réalité, une mesure quantifiant la disparité des normales associées doit être utilisée. Pratiquement, une mesure triviale de variance n'est pas suffisamment discriminante, pour le cas où les caméras et les points de la scène sont regroupés dans une direction générale autour de l'objet à reconstruire, ce qui est généralement le cas dû aux contraintes d'acquisition et de mise en correspondance. En effet, dans ce cas les normales associées aux voxels présentent une disparité qui diminue avec l'éloignement aux caméras, ne permettant donc pas une sélection des voxels corrects par simple seuillage. Nous présentons donc des mesures de disparité heuristiques permettant de prendre en compte cette atténuation de la disparité avec la profondeur.



Information pour une vue et une correspondance Étant donné la correspondance entre un pixel $\mathbf{p}$ et un point de la scène $\mathbf{Q}$, l'information disponible est la famille $\{\mathbf{P}, \mathbf{n}\}$ de surfaces et d'orientations vérifiant cette correspondance



Informations pour plusieurs vues Deux positions et orientations possibles $\mathbf{P_1}$ et $\mathbf{P_2}$ sont représentées pour le pixel $\mathbf{p}$, obtenues à partir de la correspondance entre $\mathbf{Q}$ et $\mathbf{p}$. Étant donné une deuxième vue de ces deux points, on observe que la normale bleue associée à $\mathbf{P_2}$ étant donnée la correspondance entre $\mathbf{p'_2}$ et $\mathbf{Q'_2}$ n'est pas consistante avec la normale obtenue avec la première caméra. Pareillement, la normale verte associée à $\mathbf{P_1}$ étant donnée la correspondance entre $\mathbf{p'_1}$ et $\mathbf{Q'_1}$ est identique à la normale obtenue avec la première caméra. La seule solution possible pour la surface pour le pixel $\mathbf{p}$ est donc située au point $\mathbf{P}_1$.

Résultats

La validation de cette méthode de reconstruction à été faite en deux parties, premièrement avec des données synthétiques, puis sur un objet spéculaire réel. La validation synthétique montre que des résultats quantitativement et qualitativement corrects peuvent être obtenus, avec des voxels extraits proches de la surface d'origine, et sans voxels aberrants tant que des seuils "raisonnables" sur la disparité sont appliqués. La validation sur un objet réel a par ailleurs montré que la méthode est applicable en pratique.

Discussion

Ce chapitre a présenté une extension aux méthodes de space carving pour y incorporer l'habilité de traiter les surfaces purement spéculaires. Bien que donnant des résultats

satisfaisants, cette méthode est d'une applicabilité limitée, du fait du nombre d'images nécessaires et surtout le besoin d'une mise en correspondance explicite entre pixels et points de la scène, ce qui constituait l'avantage principal des méthodes de space carving.

Triangulation

Pour relâcher certaines des contraintes inhérentes à la méthode précédente, ce chapitre présente une contrainte géométrique simple permettant la reconstruction de points d'une surface spéculaire, indépendamment les uns des autres. Les pré-requis sont l'utilisation d'une caméra dont le calibrage interne est connu, c'est à dire dont on connaît pour un pixel donné, dans un repère attaché à la caméra, le rayon de projection associé. Le principe, présenté sur la figure page 19, consiste en la triangulation des deux segments de droite constituant la réflection :

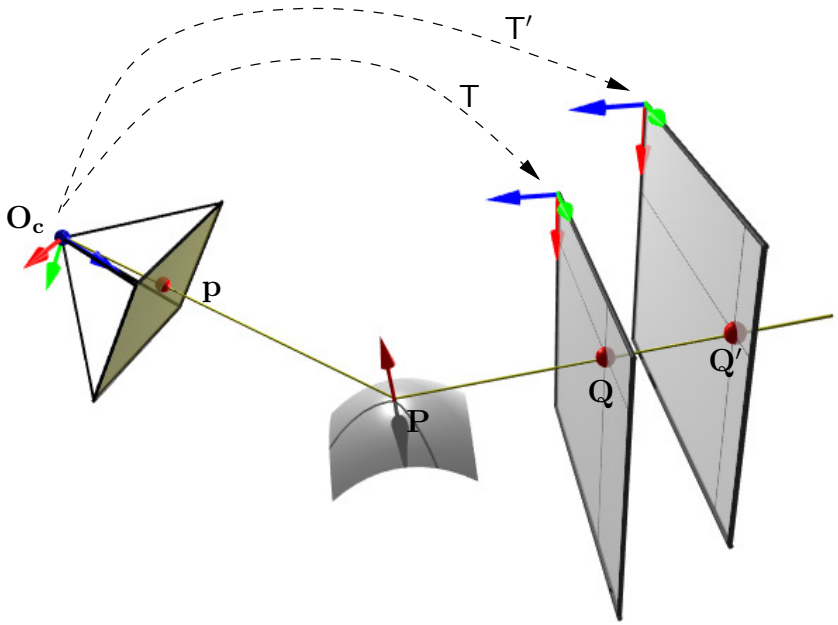
- le premier segment de droite est défini par le calibrage interne de la caméra. En pratique, le modèle de caméra utilisé est un modèle sténopé, et donc le segment de droite est défini par la droite passant par le centre optique de la caméra et le pixel.
- le deuxième segment est défini par les correspondances provenant d'un plan placé à deux positions distinctes connues.

Pour chaque triplet $\{\mathbf{p}, \mathbf{Q}, \mathbf{Q}'\}$, le point du miroir correspondant $\mathbf{P}$ est trivialement obtenu comme l'intersection des deux droites $(\mathbf{O}, \mathbf{p})$ et $(\mathsf{T}\mathbf{Q}, \mathsf{T}'\mathbf{Q}')$, et de plus la normale à la surface en $\mathbf{P}$ est définie comme la bissectrice de ces deux droites.

Résultats

La validation des résultats a été faite sur des scènes comportant plusieurs objets spéculaires, dont un exemple est montré sur la figure 3.3 page 89. Pour quantifier la précision des résultats, nous avons ajusté un plan aux points correspondants aux parties planaires de la scène, puis relevé les distances de chaque point extrait au plan ajusté. Les résultats obtenus se sont révélés d'une précision largement suffisante pour un grand nombre d'applications, avec 98% des points reconstruits situés a moins de 2 dixièmes de millimètres du plan ajusté.

Une autre validation sur une pièce de grande taille à été effectuée sans modifications sur un parebrise de voiture, permettant l'obtention d'un modèle de plusieurs centaines de milliers de points indépendants, mais qui n'a pas pu être validé quantitativement faute de vérité terrain.



Principe de la reconstruction par triangulation La mise en correspondance d'un pixel avec les points provenant d'un plan placé à deux positions différentes et connues est suffisante pour uniquement reconstruire le point du miroir correspondant.

Conclusion

Ce chapitre a présenté une méthode de reconstruction simple permettant la reconstruction de tout type de surfaces spéculaires avec des contraintes et une précision permettant une application au contrôle qualité industriel. Si la méthode en elle même est directe, ses pré-requis ne sont pas triviaux en pratique, notamment l'obtention de la pose des plans aux deux positions différentes, et la mise en correspondance dense entre pixels et points des plans. La partie suivante propose des solutions allant dans ce sens.

Pose indirecte

Une hypothèse pratique faite par toutes les méthodes de reconstruction de surfaces réfléchissantes est la connaissance de correspondances entre les pixels de l'image et leur origine connue dans la scène. Le calibrage de la scène, c'est à dire en pratique l'obtention des coordonnées dans un repère connu de points caractéristiques d'un objet de calibrage, est un problème non trivial en dehors de certains cas contraints. En effet, l'objet de calibrage doit être placé de telle façon que sa réflection soit visible dans l'objet spéculaire, limitant donc ses positions possibles, et de plus, en général, la caméra est focalisée sur l'objet spéculaire afin d'obtenir le maximum de résolution. La solution directe consistant à imposer une vue directe d'un objet de calibrage, comme par exemple sur la figure 4.2 page 98, est donc peu adaptée à une méthode de reconstruction flexible, ne serait-ce de part la perte de résolution sur l'objet à reconstruire lui-même. Ne pas imposer une vue directe de l'objet de calibrage apporte donc les avantages suivants :

- Toute la résolution du capteur est utilisée pour la surface réfléchissante elle même, ce qui est le cœur du problème.
- La caméra et l'objet de calibrage peuvent être placés au mieux, permettant de maximiser la surface de l'objet reconstruite à chaque acquisition.

Approches "directes"

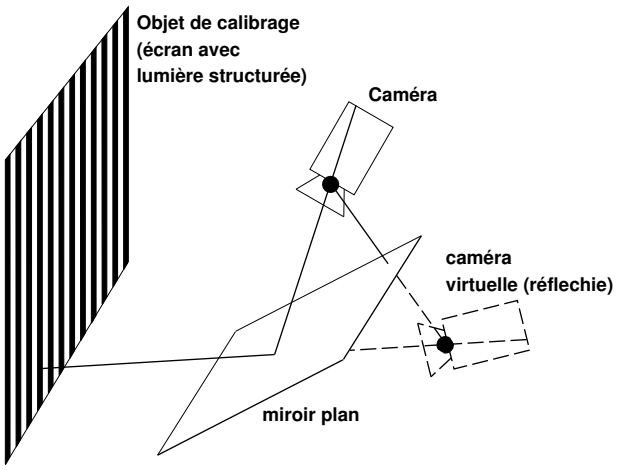
Réflection par un plan connu Une solution évidente est l'utilisation d'un miroir plan dont on connaît la pose, placé de telle façon qu'il reflète l'objet de calibrage. Des solutions pratiques pour obtenir la pose du plan incluent l'utilisation de marqueurs sur la surface, ou l'utilisation de disques de diamètres connus (plateaux de disque dur par exemple) puis un calibrage à partir d'ellipses.

Hand-Eye Dans le cas où l'on souhaite obtenir la pose d'un objet rigidement attaché à une caméra, comme par exemple sur la figure 2.15 page 81, une formulation analogue au

problème classique du Hand-eye peut être utilisée. Cette méthode nécessite par contre des acquisitions supplémentaires qui doivent être faites hors ligne.

Utilisation de la réflection par des plans inconnus

Cette partie s'intéresse à la pose d'un objet de calibrage en utilisant directement les images utilisées pour la reconstruction, lorsque l'objet à reconstruire comporte des parties que l'on sait planaires. Nous détaillons une méthodologie permettant de déterminer la pose de ces parties planaires, dès lors que l'on en a trois ou plus à disposition.



Pose réfléchie La pose estimée au travers la réflection d'un miroir plan est la pose d'une caméra virtuelle qui est la réflection au travers du miroir plan de la caméra originale.

Principe Le principe général est le suivant :

- A partir des correspondances réfléchies au travers des parties planaires, utiliser une méthode classique de pose afin d'obtenir les poses des caméras virtuelles.
- Utiliser la contrainte qu'avec les poses "correctes" des miroirs plan, les réflections des caméras virtuelles seront identiques entre elles.
  - en utilisant chaque paire de miroirs plans pour chacune des composantes rotationelle.
  - puis toutes les composantes translationnelles.
- Utiliser les poses des miroirs plans pour obtenir la pose de l'objet de calibrage (avec de plus un ajustement de faisceaux).

Résultats La validation a été faite en comparant des poses directes avec les poses obtenues par la méthode présentée, pour un exemple illustré sur la figure 4.6 page 109.

Pour un objet placé à environ 70 cm de la caméra, la différence obtenue est de l'ordre de 1 cm pour la position et 2 degrés dans l'orientation.

Le tenseur spéculaire

Ce chapitre présente une contrainte théorique permettant d'obtenir la pose d'un objet de calibrage au travers la réflection d'un objet quelconque, et donc après application de la méthode par reconstruction par triangulation, la reconstruction d'objets spéculaires en environnement non calibré. On suppose une caméra de type sténopé dont les paramètres internes sont connus observant la réflection dans l'objet à reconstruire d'un plan placé à deux positions différentes, c'est à dire un scénario identique à celui de la reconstruction par triangulation. La contrainte que nous utilisons repose sur le fait qu'avec les poses correctes pour les deux positions du plan, les droites formées par le rayon de projection issu de la caméra d'une part, et les correspondances provenant des deux plans d'autre part, doivent s'intersecter par construction. Cette contrainte peut s'interpréter par la coplanarité des quatre points en question, ce qui algébriquement correspond à la nullité du déterminant de ces quatre points. A partir de 26 ou plus des ces correspondances, on montre qu'il est possible de linéairement extraire les paramètres des poses des deux plans.

Des expériences par simulation ont malheureusement montré l'extrême instabilité du système, rendant la méthode inapplicable dans des situations rééles.

Conclusion et perspectives

Cette thèse s'est intéressée au problème de la reconstruction de surfaces réfléchisantes à partir d'images, un domaine relativement peu exploré dans le monde de la vision par ordinateur, ce qui implique qu'en général ces surfaces étaient évitées ou considérées comme du bruit. Nous avons présenté deux méthodes différentes permettant la reconstruction de ce type de surfaces, dont une permettant d'obtenir une précision suffisante pour un grand nombre d'applications, et ce sans recourir à la supposition classique de continuité ou dérivabilité.

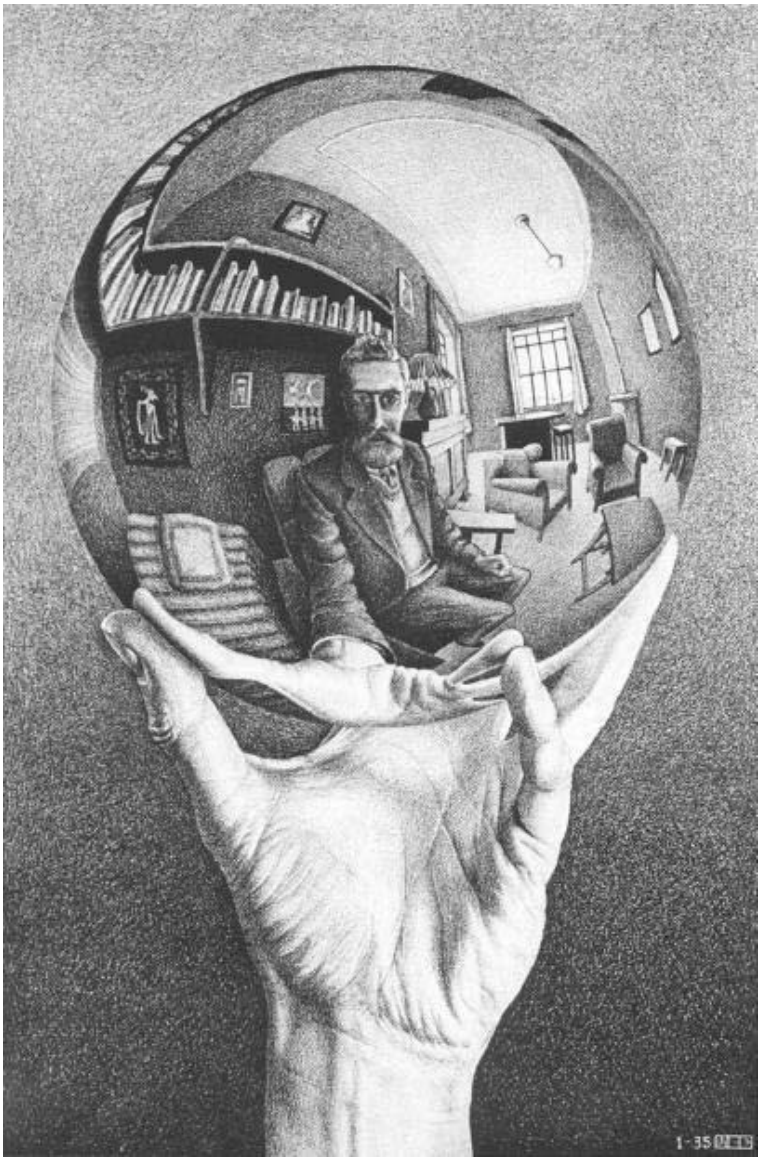
La méthode par space carving étend les méthode bien connues appliquées aux surfaces mattes. Malheureusement au vu des applications possibles de la reconstruction de surfaces réfléchissantes, nous pensons qu'elle est peut-être trop complexe à mettre en œuvre pour une utilisation "ludique" (réalité virtuelle par exemple), et pas assez précise pour des utilisations industrielles de type contrôle qualité.

La méthode par triangulation nous a permis d'obtenir des résultats très convaincants, avec une mise en œuvre simple et peu contraignante. Dans un environnement plus contraint dans le cadre d'une application industrielle par exemple, la précision peut être

encore améliorée, en modélisant précisément la caméra et en connaissant exactement la position des plans utilisés.

Finalement, les méthodes présentées permettant d'obtenir la pose d'un objet hors du champ de vue d'une caméra répondent à un problème pratique commun à toutes les méthodes de reconstruction de surfaces réfléchissantes, et peuvent aussi servir pour l'obtention de pose de systèmes multi-caméras dans le cas ou celles-ci n'ont pas de champ de vue en commun. Finalement, la partie présentant la contrainte permettant d'obtenir la pose d'un objet à travers la réflection d'un objet spéculaire quelconque aurait permis l'élaboration d'une méthode flexible et précise de reconstruction. L'instabilité observée révèle peut-être la limite en termes de flexibilité des scénarios possibles pour l'obtention de la géométrie des surfaces réfléchissantes.

Un problème pratique à améliorer reste celui de la mise en correspondance automatique entre points de la scène et pixels de l'image, ayant à la fois la précision suffisante et la rapidité d'exécution pour des scénarios industriels. Enfin, des méthodes ne requérant pas un environnement calibré est un point de départ obligatoire avant d'obtenir des algorithmes de reconstruction génériques flexibles, permettant l'obtention de modèles de tout type de surfaces. Ceci passera probablement par l'intégration d'autres sources d'information, notamment des a priori sur la surface ou des contours occultants.



Escher - Hand with Reflecting Sphere 1935, Lithograph

SPECULAR SURFACE RECONSTRUCTION FROM IMAGES

Contents

List of Figures

Notation

Throughout this thesis, we will use the following notation to express mathematical and geometrical entities:

- a scalar is represented by italic letters, for example s.
- a vector or a point in space will be represented by bold letters, for example $\mathbf{v}$. A null vector will be represented as $\mathbf{0}$, or $\mathbf{0}_n$ when specifying its dimension.
- a matrix will be represented with capital sans-serif letters, for example M. The identity matrix is represented as $\mathtt{I}$.
- the $\sim$ sign will represent equality up to scale, when working with projective geometry.
- $[\mathbf{v}]_\times$ is the matrix that represents the cross-product by vector $\mathbf{v}$. We therefore have $\forall \mathbf{u} \in \mathbb{R}^3, [\mathbf{v}]_\times \mathbf{u} = \mathbf{v} \times \mathbf{u}$:

$$[\mathbf{v}]_\times = \begin{bmatrix} x \\ y \\ z \end{bmatrix}_\times = \begin{pmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{pmatrix}$$

- M^T is the transpose of matrix M.

Introduction

For the last 30 years, the computer vision community has striven for an ideal object reconstruction approach from images. Whereas the human visual system instantly recognizes objects, computes a 3 dimensional structure of virtually any environment, and obtains reflectance cues from most types of surfaces, building an artificial system that mimics this behavior remains a very open problem.

The advantages of such a system would be tremendous. Compared to physical models, numerical models can be inspected at any resolution, displayed in any environment, broadcast over communication wires, indexed in databases, stored efficiently, manipulated in CAD applications . . . the possibilities are infinite. The fields of application concerned that come to mind include archeology, preservation of cultural heritage, surgery, assistance for the disabled, entertainment, communication, robotics, industrial design and quality control, e-commerce, virtual sight-seeing, etc. . .

Systems that are able to produce these numerical models exist, but require expensive and complicated equipment. Contact probes placed on the end of a robotic arm can produce very precise measurements, at the expense of having to physically interact with the object, and goniometers can be used to capture a full reflectance function of a given type of surface. While these techniques are general, they remain time consuming. Laser scanners also rapidly allow the capture of a complete 3D model, with the notable exception, especially for the subject of this thesis, of highly reflective surfaces. Obtaining such models using only images, on the other hand, would be highly appealing, as:

- Appliances that create images from observation are common, cheap and non-invasive.
- The problem seems tractable, since that is precisely what the human visual system does.

The human being integrates a large variety of cues provided by physiological and psychological mechanisms [Oko76]. The physiological cues can be binocular, such as parallax and convergence, or monocular in the case of movement parallax (*i.e.* perception of 3D shape with only one eye by slightly moving the head) or accommodation

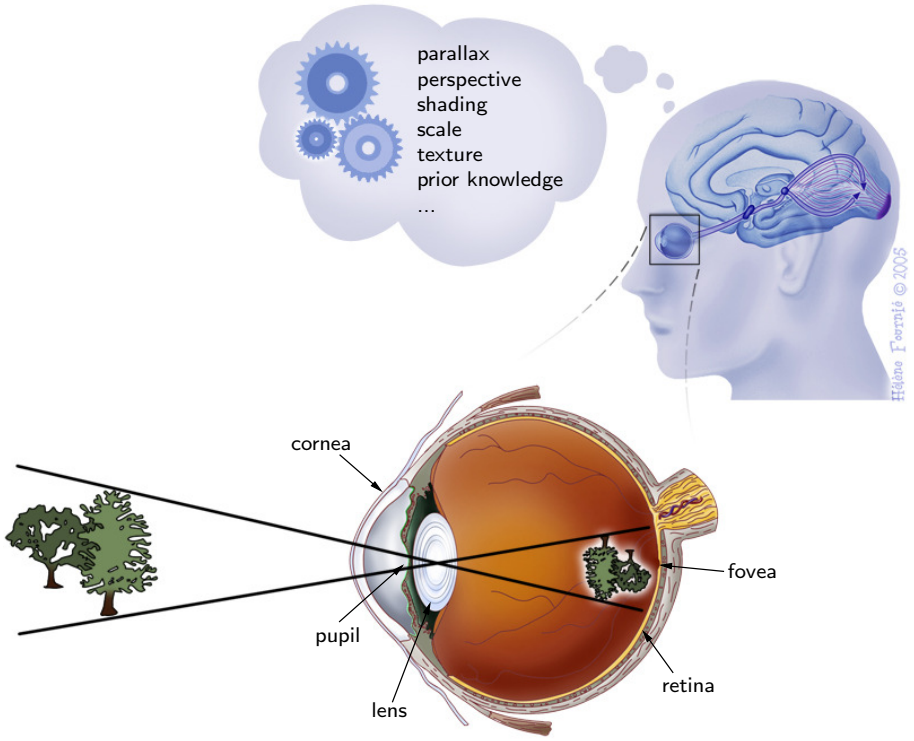


Figure 1: **The human visual system**

(*i.e.* muscle tension due to focusing distance). The psychological cues are all monocular, and include scale (for known objects), linear perspective (intersection point of parallel lines), texture gradient, and shading and shadows. Another capital source of information is prior knowledge: having recognized the shape of a human body, we *know* that we will find a face and hair towards the top, and two legs down on the ground.

The complexity of the constraints provided by these cues is mind boggling, and in the mean time is too important to allow them to be fully integrated in an artificial system. However we can find strong analogies between the development of an artificial vision system and the human visual system; the newborn baby already possesses a powerful acquisition system (the eyes), but has not learned to use them. He only recognizes shapes as the intersection of lines formed by the border between light and dark regions. Ability to track objects, and estimation of depth, develop after three months of constant training. Precise distance judgments rarely occur before the age of one. And full coordination with the muscular system is acquired in the school years. From this, we can infer that the state of the art of artificial vision is in its first stammerings.

Whereas the human being integrates at each instant *all* the cues cited above, the complexity of these cues bind researchers to work on only a subset of the available information. For example, most stereoscopic algorithms require Lambertian reflectance and ignore any specular contribution, reflectance estimation techniques assume known shape and lighting, and are limited to a finite number of possible reflectances, recognition algorithms are tailored for only a (tiny) number of recognizable classes ...

Having described what we believe is an ideal system, we now focus on the subset that concerns this thesis: obtaining object or surface structure from images. Given an object to reconstruct, we therefore wish to obtain a numerical representation of how it occupies space, in other words the coordinates in 3 dimensions of the points that constitute it.

Most of the results on the subject have been obtained by making drastic assumptions on surface reflectance, by supposing that the brightness observed on a point of a surface is invariant to the locus of observation. Surfaces that verify this constraint to some extent include plaster, paper, clothes or unvarnished wood. For methods using feature matching such as stereo, voxel coloring, or structure from motion, the reason for this assumption is that surface patches imaged from different viewpoints must be identified and matched in these images. The geometrical information then given by these matches allows geometrical constraints to be applied to the structure (namely position) of the surface patch. For shape from shading algorithms, assuming matte and uniform reflectance allows surface orientation to be determined using only brightness and the position of the illuminant.

However, most surfaces surrounding us are far from satisfying this constraint. Techniques that suppose matte shading all produce false if not aberrant results if their input data does not fit to this model. This is the case for purely specular surfaces:

- Methods relying on the matching of surface patches will usually reconstruct points behind or in front of the correct surface, depending upon whether the surface is convex or concave.

- In shape from silhouette methods, the difficulty is the extraction of the silhouette at grazing angles, as the texture of the specular surface and background are identical (see figure 2(d) for an illustration).

- The energy output by a laser scanner will be reflected in the mirror direction and never reach the associated captor.

As such, most techniques usually discard specularities as noise, therefore producing object models that contain holes. Another approach is to use a constrained setup where the positions of light sources, cameras and objects are carefully and manually chosen so that these specularities do not occur or at least do not occur on important locations.

Motivations

What use is there in reconstructing specular surfaces from images? Apart from the fact that this was probably the most common reaction when explaining my work to acquaintances, the question calls for an answer.

It would not be an exageration to say that perfect mirrors or surfaces with a sharp specular component are very uncommon, therefore algorithms or methods that focus only on these kinds of surfaces may seem peculiar. However, there is a range of applications where such a problem directly arises:

- The automobile industry chain produces large quantities of bodyworks and windshields, all of which have a non negligible specular component (figure 2(a)). At present, the quality control of such parts at the end of the production chain is obtained by contact probes. This kind of measurement is cumbersome and time consuming, and therefore can only be performed on a finite number of samples each day. Of course, a measurement concluding to a faulty part implies that all the previous production be at the best individually verified, or at the worst discarded. Non-invasive real-time inspection is becoming widespread for the inspection of the micro-structure of such parts (*i.e.* detecting local irregularities in surface curvature), but a near real-time system able to compute a global model that can then be compared to a reference is highly desirable.

- Industrial metrology is a similarly demanding sector, for the quality control and inspection of shiny tubes, jewelry, ceramics... (figure 2(b)).

- Optics inspection, as most cameras and telescopes use mirrors as well as lenses to transport light to the desired location (figure 2(d)).
- In the surgical field, most of the inner tissues of the living body are highly specular. For computer assisted interventions, methods that are able to treat such specular surfaces will be required if a 3D model is needed (figure 2(f)).
- Cultural heritage preservation, where some objects tend to be highly specular (figure 2(e)).
- Virtual reality, where reconstructed models should also incorporate possible mirror surfaces (figure 2(c)).

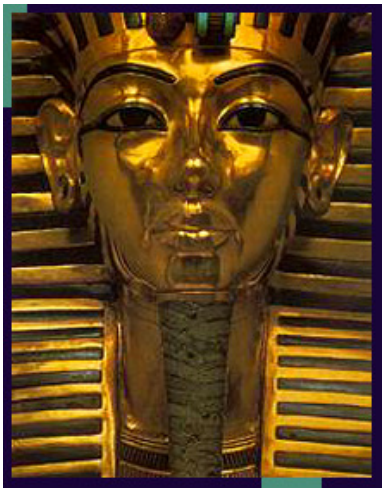
But the heart of the problem remains elsewhere. As much as traditional computer vision algorithms are incomplete as they do not treat specularities, whatever techniques treating only these specularities are equally as deceptive. However, being more complicated to treat and less widespread, such types of surfaces have received relatively little attention from the computer vision community. Work on purely specular surfaces must therefore be seen as a step towards more general algorithms that can incorporate multiple visual cues, rather than a goal in itself.

(a) Automobile bodywork inspection

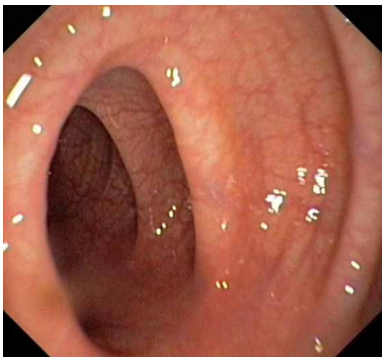
(b) Industrial inspection

(c) Virtual reality

(d) Optics inspection



(e) Cultural heritage



(f) Surgery

Figure 2: **Applications for highly specular surfaces**

Presentation of the Problem

When working with specular surfaces, we must employ a specific vocabulary, as new concepts are introduced with respect to more typical geometry. These new concepts arise from the fact that a specular surface usually has a very rich apparent texture, that corresponds to the environment it reflects towards the viewer, and not to a texture that is attached to the surface. This implies that a feature a viewer sees on a specular surface is carried by a ray of light that hasn't traveled in a straight line as if it were seen directly, but by two rays: one from the feature to a point on the specular surface, and one from that point to the viewer. This point on the specular surface is physically defined by its 3D coordinates, but not by its appearance, as the latter will change depending on from where it is viewed. To distinguish between these different types of feature points, we will use the following vocabulary:

- A point on a non-reflecting surface, usually characterized by a specific texture that describes a feature, will be referred to as a *scene* point. Such a scene point will be called *calibrated* if its 3D location is known, or *unknown* if not.
- A physical point belonging to a mirror surface will be called a *mirror* point. This point has a viewpoint dependent texture, which means that throughout different views, identical texture patches do not correspond to identical mirror points. A feature coming from a scene point and reflected by a specular surface at the mirror point will be called a *virtual* feature.
- The projection of a point onto the image plane of a camera corresponds to a *pixel*, after some kind of quantization is applied due to digitization. A pixel has a physical 3D location, but the relevant information it provides is in fact the projection ray to which it corresponds.

The typical path that is taken by a ray of light, shown on figure 3 when working on specular surfaces is therefore:

All the difficulty of treating specular rather than traditional objects arises from this path, as the geometrical information given by a pixel and a scene point corresponds to *two* lines (the rays of light) rather than one. Even in the very constrained case were a calibrated camera observes the reflection of a known scene point, there is a 1-dimensional family of surfaces along the corresponding projection ray that could have produced this observation, as shown on figure 4. The aim of specular surface reconstruction techniques is thus to find additional constraints that make the problem tractable. Note that additional complexity can be introduced if we consider interreflection: in that case, the

path followed by a ray of light will consist of 3 or more such rays. We will therefore always ignore intereflections.

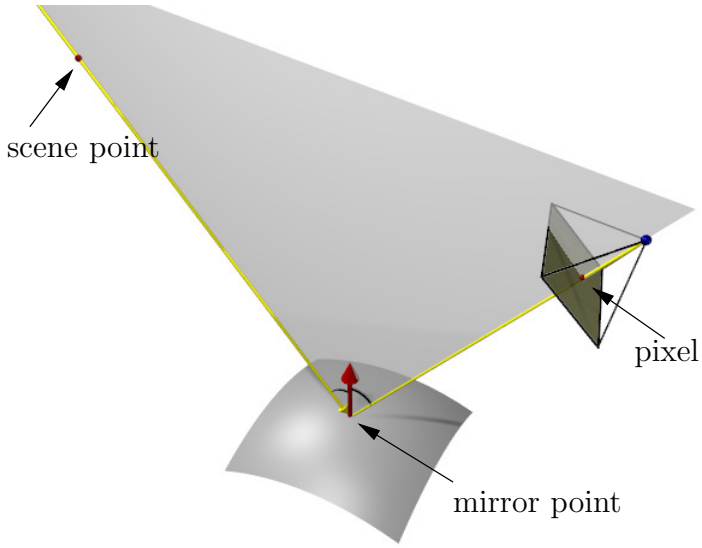


Figure 3: **Vocabulary: mirror points, scene points and pixels**

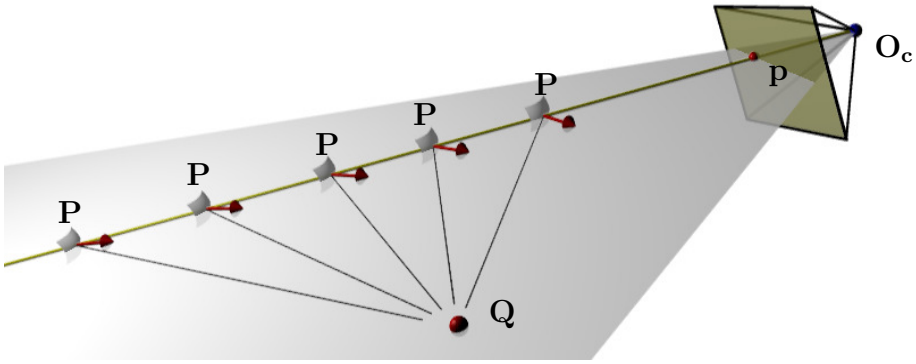


Figure 4: **1-Dimensional family of possible configurations.** Given the correspondance between a pixel $\mathbf{p}$ and a scene point $\mathbf{Q}$, there is a one dimensional family of possible configurations for the position of the mirror point $\mathbf{P}$ along the projection corresponding to $\mathbf{p}$.

Contributions and Document Outline

The contributions of this thesis are the following:

- In chapter 2, a specular surface reconstruction method in the spirit of space carving methods will be presented. Rather than using photometric consistency, we use a geometrical measure that represents the consistency of surface orientation.
- A method to compute a dense point cloud of a specular surface, using the reflection of points from two planes is given in chapter 3. The points are all independent from one another. The foundation of the method is very similar to that of generic camera calibration.

These two methods allow a dense reconstruction of a specular surface, without assuming surface continuity or smoothness, as usually the case in the state of the art.

An often overlooked but nevertheless essential practical problem in most specular surface reconstruction technique is the obtention of calibrated scene points as seen from the imaging devices (or conversely the pose of these imaging devices in a global reference frame). The problem resides in the fact that cameras have a field of view in which it is often difficult to incorporate the specular object *and* a calibration object or marker points. The final contributions of this thesis, allowing more flexible setups in this case, are:

- Practical and theoretical methods for obtaining the pose of a calibration object when it is not in direct view of a pinhole camera, in chapter 4.

1.

Specular Surfaces and Computer Vision

This chapter introduces the general concepts of computer vision which will be used in the rest of this thesis. It includes an introduction to surface reflectance, projective geometry and the commonly used model of a camera. Then follows a state of the art of the most relevant work concerning specular surfaces.

1.1 Image Formation

1.1.1 Light and Surfaces

The appearance of surfaces varies greatly from one surface to another, from one illumination condition to another, and from one point of view to another. These variations must typically be taken into account by any computer vision algorithm, if it aims to achieve at least some kind of robustness. Algorithms able to treat any kind of surface, and/or under any illumination, and/or from any viewpoint have yet to be researched, as for the meanwhile, even the most powerful ones apply to only a subset of all possible appearances.

Whilst the study of light and the way it interacts with surfaces isn't the central subject of this thesis, the mere fact that we are concerned with specular surfaces justifies our spending a little time exploring the subject. The object of this section is to formalize the appearance of a surface patch when observed from a specific direction.

The distribution of light in space is called *radiance*. From [SP94], radiance is

> *the amount of energy travelling at some point in a specified direction, per unit time, per unit area perpendicular to the direction of travel, per unit solid angle.*

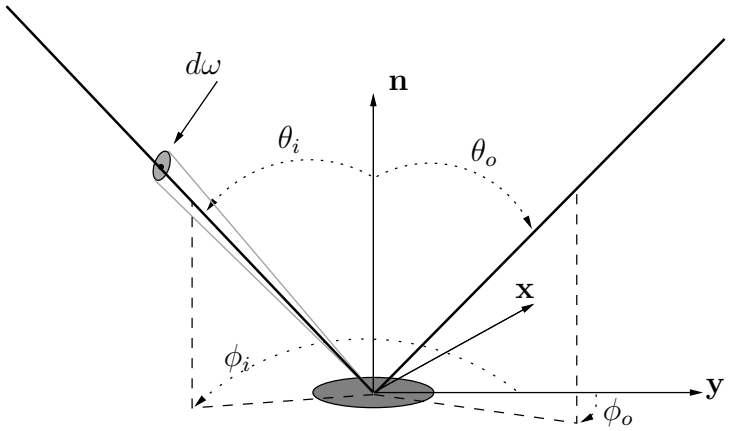


Figure 1.1: **Light reflectance geometry** We place ourselves in the reference frame aligned on the surface normal $\mathbf{n}$. The orientation of $\mathbf{x}$ and $\mathbf{y}$ around $\mathbf{n}$ is arbitrary, but is usually chosen as to be aligned with the world reference frame.

For our study, we are interested in the interaction of light when it hits the surface of an object. The quantity that interests us is called *irradiance*, defined as

> *the amount of incident energy per unit time per unit surface.*

The irradiance at a surface point $\mathbf{x}$ originating from radiance $L_i(\mathbf{x}, \theta_i, \phi_i)$ of differential solid angle $d\omega$ is calculated as $L_i(\mathbf{x}, \theta_i, \phi_i) \cos(\theta_i) d\omega$, where $\cos(\theta_i)$ is a foreshortening factor. The total energy incident at point $\mathbf{x}$ is therefore the integral of the irradiance over the point's hemisphere.

Irradiance and radiance are the quantities that we use to describe the interaction of light with a surface. We first make a few assumptions concerning this surface:

- the surface does not emit any light
- the radiance exiting at point $\mathbf{x}$ is due only to the irradiance at point $\mathbf{x}$, *i.e.* there is no scattering of light under the surface of the object.
- the wavelength of incident light is not modified when interacting with the surface. This allows us to ignore wavelengths in the following definitions.

The *bidirectional reflectance distribution function*, widely referred to as *BRDF* and noted ρ_{brdf}, is the most common model describing this interaction, and is defined as

> *the ratio of the radiance in the outgoing direction to the incoming radiance.*

and as such measures the fraction of light from a given incident direction which is reflected to a given outgoing direction.

By denoting outgoing radiance as $L_o(\mathbf{x}, \theta_o, \phi_o)$, the BRDF at point $\mathbf{x}$ is therefore

$$\rho_{brdf}(\mathbf{x}, \theta_i, \phi_i, \theta_o, \phi_o) = \frac{L_o(\mathbf{x}, \theta_o, \phi_o)}{L_i(\mathbf{x}, \theta_i, \phi_i) \cos(\theta_i) d\omega}$$

Lambertian Surfaces

Most computer vision algorithms assume a Lambertian reflectance model for the surfaces they analyse. In the Lambertian model, an incoming ray of light is equally reflected in all directions by the surface, or in other words, the BRDF is a constant function. This leads to a viewpoint independent model of appearance of the surface, which is very practical for automated matching algorithms commonly used in computer vision: any given point on a fixed Lambertian surface will have the same color and intensity, independently of where it was seen from.

$$\rho_{brdf}(\phi_i, \theta_i, \phi_o, \theta_o) = \frac{\rho_d}{\pi}$$

where ρ_d is called *diffuse reflectance* or *albedo*.

Whereas this is the most commonly used surface appearance model in computer vision algorithms, examples of surfaces that verify its constraints exactly are very rare, if not impossible to find. This explains the fact that highlights are usually discarded by computer vision algorithms, as they correspond to samples of the BRDF that are not modeled by the Lambertian model.

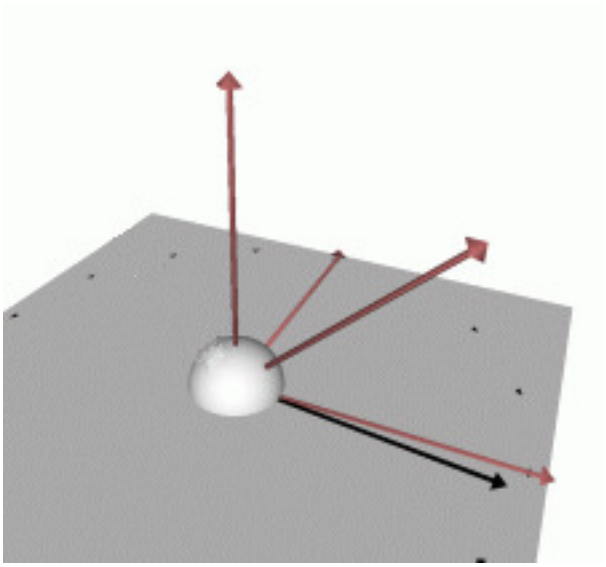


Figure 1.2: **Lambertian BRDF** For a given incident illumination, the observed brightness of the surface is independent of the viewpoint.

Perfect Mirrors

Oppositely to Lambertian surfaces, the perfect mirror is a surface that reflects all incoming energy in the corresponding specular direction. Its BRDF is therefore 0 except if the incoming and outgoing directions are symmetric with respect to the surface normal.

$$\rho_{brdf}(\phi_i, \theta_i, \phi_o, \theta_o) = \delta(\cos\theta_i - \cos\theta_o)\delta(\phi_i - (\phi_o \pm \pi)) \tag{1.1}$$

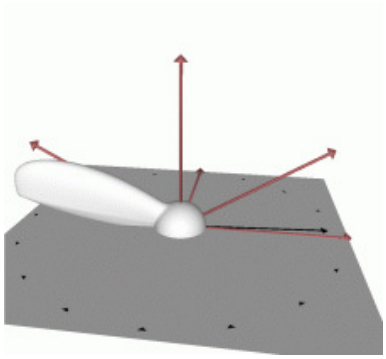
where $\delta()$ represents a dirac function.

Examples of such surfaces are uncommon in typical setups and environments, but the understanding of the geometry inherent to specular reflection is a prerequisite to algorithms able to treat hybrid surfaces.

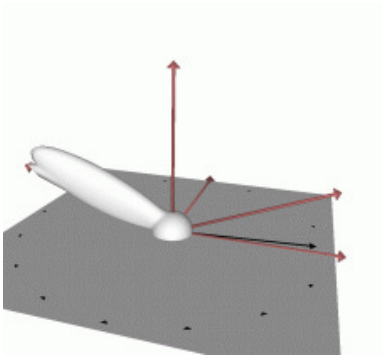
Hybrid Surfaces

The vast majority of surfaces are neither perfectly Lambertian nor perfect mirrors. Many parametric models describing such BRDFs exist, some combining Lambertian reflectance with a specular lobe, others also treating retro-reflection. Figure 1.3 shows some examples.

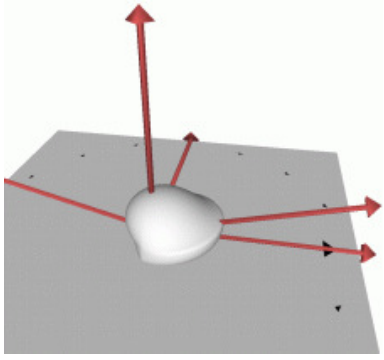
Computer vision algorithms able to treat surfaces with arbitrary BRDFs have yet to be invented. An approach to this can be found in [MKZB01] which exploits Helmholtz stereopsis in a very constrained setup.



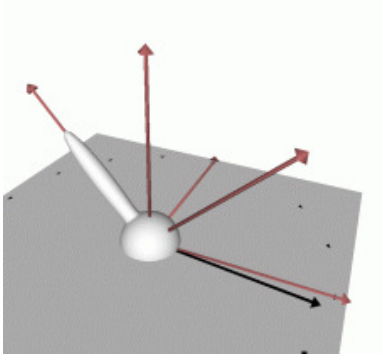
(a) Cook Torrance Specular Microfacet Model [CT82]



(b) He-Torrance Comprehensive Analytic Model [HTFG91]



(c) Oren-Nayar Diffuse Microfacet Model [ON95a]



(d) Phong Model [Pho75]

Figure 1.3: **Parametric Reflectance Models** Different parametric models proposed to describe the interaction of light with surfaces.

1.1.2 Geometric Constraints

Most of the concepts used in computer vision derive from the relation between 3D objects viewed and 2D images we have of them. The nature of such a relation is based on projective geometry, of which we will present the concepts that are required for this thesis. This section is by no means exhaustive, as its content is not central to the goal at hand. For a detailed reference, the reader is invited to refer to [HZ00],[Pol00].

Projective Geometry

A n-dimensional projective space $\mathcal{P}^n$ is composed of elements of size $(n+1)$ expressed in homogeneous coordinates. For example an element $\mathbf{x}$ is represented in $\mathcal{P}^n$ as $(x_0, x_1, \ldots, x_n)$, with the constraint that $\mathbf{x} \neq \mathbf{0}_{n+1}$. Two representations $\mathbf{x}$ and $\mathbf{y}$ of $\mathcal{P}^n$ correspond to the same element if $\exists \lambda \neq 0 \in \mathbb{R} : \mathbf{x} = \lambda \mathbf{y}$, which we also denote by $\mathbf{x} \sim \mathbf{y}$.

We make no distinction between objects, points and vectors of $\mathcal{P}^n$, as they are all represented as a 0-dimensional element. The dual to such points are called *hyperplanes*, and are $(n-1)$-dimensional subspaces of $\mathcal{P}^n$. For example in $\mathcal{P}^2$ points and lines are dual, as a line is defined by the joins of two points and a point is defined by the intersection of two lines. In $\mathcal{P}^3$, points and planes are dual.

In $\mathcal{P}^n$, a point $\mathbf{x}$ belongs to a hyperplane π *iff.*

$$\pi^{\mathsf{T}} \mathbf{x} = 0$$

of which the dual is

$$\mathbf{x}^{\mathsf{T}} \pi = 0$$

An invertible linear mapping of $\mathcal{P}^n \rightarrow \mathcal{P}^n$ is called a *projective transformation* (also referred to as a *colineation* or *homography*). It is represented by a $(n+1) \times (n+1)$ matrix.

A point $\mathbf{x}$ is transformed by

$$\mathbf{x} \leftarrow \mathsf{T}\mathbf{x}$$

and a hyperplane π by

$$\pi \leftarrow \mathsf{T}^{\mathsf{T}} \pi$$

A linear mapping of $\mathcal{P}^n \rightarrow \mathcal{P}^m$ with $m < n$ is called a *projection*, and is represented by an $(m+1) \times (n+1)$ matrix. For example, the typical task of a pinhole camera, mapping a 3D scene to an image, is a projection of $\mathcal{P}^3 \rightarrow \mathcal{P}^2$ which is therefore represented as a 3×4 matrix.

Points, lines, and planes

We now place ourselves in $\mathcal{P}^3$, the projective space that is used to describe objects in 3D. This section will describe the subsets of transformations inside this space, and the representations of points, lines and planes. The elements of $\mathcal{P}^3$ of the form $(x, y, z, 0)$ are called *points at infinity*, and represent directions. As such in $\mathcal{P}^3$, parallel lines (*i.e.* with same direction) join at infinity.

Projective transformations Projective transformations in $\mathcal{P}^3$ are represented by an 4×4 invertible matrix with 15 degrees of freedom. They preserve colinearity, incidence, conjugacy, and cross ratio. The cross ratio of four colinear points $\mathbf{X_1} \dots \mathbf{X_4}$ defined as $\mathbf{X_i} = \mathbf{X_0} + \lambda_i \mathbf{X}$ is

$$\{\mathbf{X_1}, \mathbf{X_2}, \mathbf{X_3}, \mathbf{X_4}\} = \frac{\left(\frac{\lambda_1 - \lambda_3}{\lambda_1 - \lambda_4}\right)}{\left(\frac{\lambda_2 - \lambda_3}{\lambda_2 - \lambda_4}\right)}$$

Affine transformations The affine space $\mathcal{A}^3$ is a subspace of $\mathcal{P}^3$ defined as $\mathcal{P}^3$ minus the points at infinity, or in other words:

$$\mathcal{A}^3 \to \mathcal{P}^3 : (x, y, z)^\mathsf{T} \to (x, y, z, 1)^\mathsf{T}$$

A consequence of this is that parallel lines never join in $\mathcal{A}^3$.

An affine transformation is a 12-degree of freedom projective transformation that can be represented by the 4×4 invertible matrix

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Affine transformations of course also conserve conjugacy, incidence, collinearity and cross ratio, but also have the property whereby points at infinity stay at infinity.

Euclidean transformations A subset of the affine transformations in $\mathcal{P}^3$ is denoted *euclidean transformation*. It denotes the composition of a rotation R and a translation $\mathbf{t}$, and is represented by a 4×4 invertible matrix with 6 degrees of freedom of the form

$$\mathsf{T} = \begin{pmatrix} & \mathsf{R}_{3\times3} & & \mathbf{t}_3 \\ & & & \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

the inverse of this transformation is expressed as

$$\mathsf{T}^{-1} = \begin{pmatrix} & \mathsf{R}_{3\times3}^{\mathsf{T}} & & -\mathsf{R}^{\mathsf{T}}\mathbf{t}_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

A euclidean transformation is also called a *rigid displacement*, and conserves lengths and angles.

Points A point in 3D is a 3 degree of freedom vector of $\mathcal{P}^3$:

$$\mathbf{x} \sim \begin{pmatrix} x \\ y \\ z \\ t \end{pmatrix}$$

In euclidean space, when $\mathbf{x}$ is scaled such that $t = 1$, $\|(x, y, z)^{\mathsf{T}}\|$ is the distance from the point to the origin.

Lines There are different representations of lines used in the computer vision community. We do not explicitly use lines in this thesis, and hence the subject will not be discussed.

Planes A plane in 3D is the dual of a point, and is therefore also represented as a 3 degree of freedom vector

$$\Pi \sim \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

In euclidean space, when $\|(a, b, c)^{\mathsf{T}}\| = 1$, d is the distance from Π to the origin.

Cameras

This paragraph briefly describes the geometrical properties that we assume of a typical image acquisition device. Unless otherwise stated, we will consider a camera as a very general imaging system, *i.e.* as a mapping between a point in 3D and a specific line passing through it. The task of a camera is to determine the properties of this 3D line given the observation of the corresponding 3D point. Inversely, the task of a typical

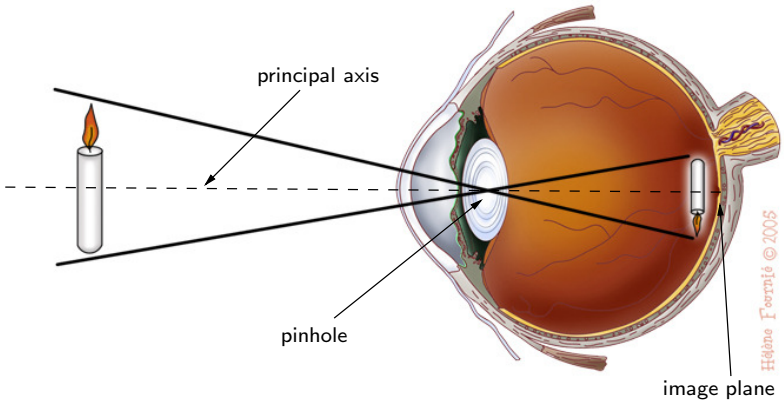


Figure 1.4: **The human eye seen as a pinhole camera**

geometrical computer vision algorithm is to compute information on the 3D point given information about its corresponding lines.

While this definition of a camera is very general, it does not resemble any imaging device we are used to encounter. Actually, the most common camera is the human eye, modelled on figure 1.4. The vast majority of cameras in the more classical sense mimic the behaviour of the human eye, in order to produce images that are visually appealing to a human observer.

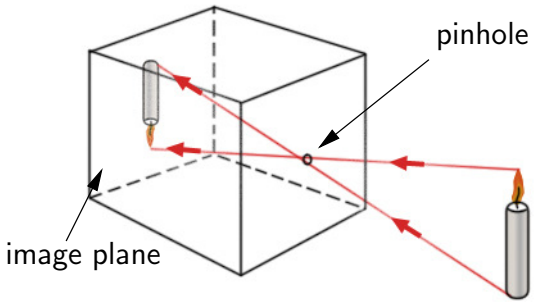


Figure 1.5: **The camera obscura is a box or a room with a hole in one wall. Light from the exterior of the box can only enter the box through this hole, thus hitting the opposite wall at a specific location, forming an upside down image of the surroundings.**

Man-made cameras were popularized during the Renaissance period by Brunelleschi,

taking the form of the *camera obscura* (figure 1.5), and several famous painters are known to have used the device to help their interpretation of objects and light. The

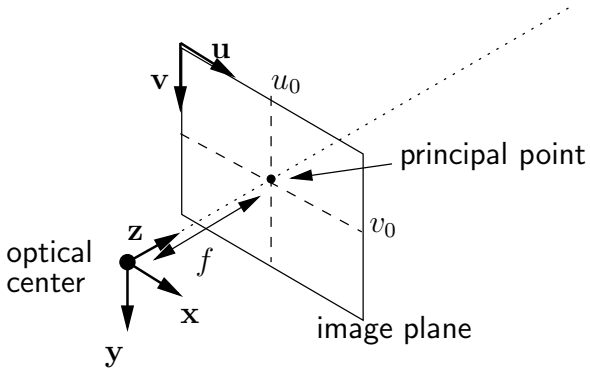


Figure 1.6: **The pinhole model**

transposition of the camera obscura to a geometrical entity is described by the *pinhole model*, whose representation and parameters are shown on figure 1.6. This type of camera samples the light rays that pass through a unique point called the optical center, and then hit the image plane, creating an image of the scene by either photochemically activating the emulsion at the surface of the film in a traditional camera, or electrically charging the sensors of a digital camera.

This type of camera can easily be modelled with a handful of parameters, which greatly reduces the computational complexity of computer vision algorithms. Although most handheld cameras produce distortions that are not accounted for in this model, it is sufficiently accurate for most applications.

The pinhole camera performs a *projection* from $\mathcal{P}^3$ to $\mathcal{P}^2$. It is therefore defined by a 3×4 projection matrix P such that a 3D point $\mathbf{Q}$ is projected onto a pixel of the image plane $\mathbf{q}$ by

$$\mathbf{q} \sim \mathrm{P}\mathbf{Q}$$

While P can be given directly to describe such a projection, it is clearer to decompose the resulting projection through the transformations between 4 reference frames, shown on figure 1.7:

1. the world reference frame $\mathcal{W}$, whose center and orientation can be fixed arbitrarily, and usually corresponds to the reference frame of a calibration object

2. the camera frame $\mathcal{C}$, whose origin is the center of projection of the camera, and whose axes are aligned with the image plane

3. the image frame $\mathcal{I}$ centered on the principal point

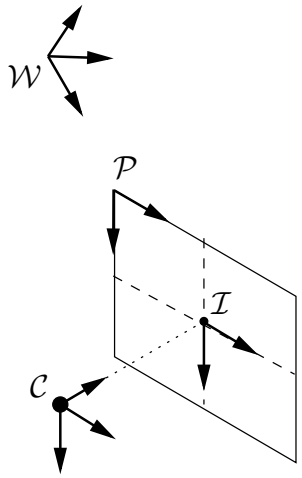


Figure 1.7: **Reference frames involved in a camera projection**

4. the pixel frame $\mathcal{P}$ which corresponds to the coordinates in pixels of the points in the image frame.

The transformations between these frames are the following:

- a euclidean transformation from the world reference coordinate system to the reference coordinate system where the camera is defined

$$\begin{pmatrix} & \mathsf{R}_{3\times 3} & & \mathbf{t}_3 \\ & & & \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- a projection from the camera coordinate system to the image plane coordinate system

$$\begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

- an affine transformation inside the image plane from image frame to pixel frame (we deliberately ignore the case of non-orthogonal image axes)

$$\begin{pmatrix} k_u & 0 & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

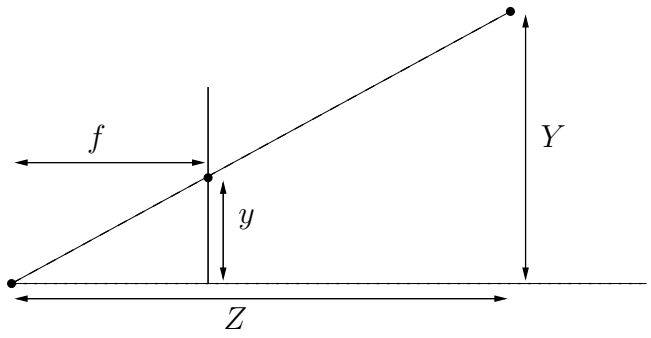


Figure 1.8: **Projection in 2 dimensions.** From Thales' law we have $f/Z = y/Y$, thus the projection of point (Y, Z) on the image plane at f is $y = fY/Z$.

The parameters used in the above transformations are the following:

- the focal length f measures the orthogonal distance from the optical center to the image plane. As seen in 2 dimensions on figure 1.8, a point with coordinates (Y, Z) is projected on point y of the image plane with the relation $y = fY/Z$

- the *principal point* (u_0, v_0), which represents the coordinates in the image plane of the orthogonal projection of the optical center. The principal point also defines the *principal direction* which is the half line starting at the optical center and passing through the principal point.

- k_u and k_v represent the inverse size of the pixels in the image plane for each of the principal directions.

The complete projection matrix is therefore:

$$\mathsf{P} = \begin{pmatrix} k_u & 0 & u_0 \\ 0 & k_v & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} & \mathsf{R}_{3\times3} & & \mathbf{t}_3 \\ & & & \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

The camera projection and affine transform in the image plane are usually concatenated in a 3×4 matrix K called *calibration matrix*:

$$\mathsf{K} = \begin{pmatrix} k_u f & 0 & u_0 & 0 \\ 0 & k_v f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

A common representation is to represent the focal length f in pixel units, by noting $\alpha_u = k_u f$ and $\alpha_v = k_v f$:

$$\mathsf{K} = \begin{pmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

The calibration matrix K obtained thus represents the 4-degree of freedom projection function of a camera. However, in most recent cameras, the image plane is composed of square pixels, therefore reducing the degrees of liberty to 3, as in this case $k_u = k_v$ (or $\alpha_u = \alpha_v$).

Given a complete projection matrix P obtained by an arbitrary calibration method, K,R and $\mathbf{t}$ can be extracted by QR decomposition [Tsa86].

Law of Reflection

The geometric constraints that we will use to derive geometrical information from observations are straightforward. To begin, we suppose, without lack of generality, that light traveling in the air can be modeled as a ray (or ideal narrow beam), and therefore can be represented by a line segment. Having assumed this, we can apply the classical laws of optics, more specifically in our case the law of reflection: when an incident ray of light hits a non-scattering (*i.e.* specular) surface it is reflected symmetrically to the normal of the surface. This formulation is equivalent to the BRDF definition of equation 1.1. Inversely, if we know the incident and reflected rays at a point of a specular surface, we can determine the orientation of the surface at that point as the bisector of these two rays.

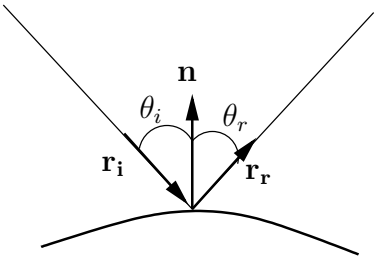


Figure 1.9: **Law of reflection** The angle θ_i between the incident ray $\mathbf{r}_i$ and the surface normal $\mathbf{n}$ is equal to the angle θ_r between the surface normal $\mathbf{n}$ and the reflected ray $\mathbf{r}_r$. The incident ray, the surface normal, and the reflected ray are coplanar.

Mathematically, these constraints are expressed by the following:

- Given an incident ray $\mathbf{r_i}$ reflected by a surface of normal $\mathbf{n}$, the reflected ray $\mathbf{r_r}$ is calculated as

$$\mathbf{r_r} = \mathbf{r_i} - 2\mathbf{n}(\mathbf{n}^\top \mathbf{r_i})$$

- Given an incident ray $\mathbf{r_i}$ and a reflected ray $\mathbf{r_r}$, which we suppose normalized, the surface normal $\mathbf{n}$ that links these two rays by a reflection is

$$\mathbf{n} = \frac{-\mathbf{r_i} + \mathbf{r_r}}{\|-\mathbf{r_i} + \mathbf{r_r}\|}$$

1.2 Previous Work

Specularities have interested researchers in the field of computer vision since the eighties, in most cases in order to be able to detect and ignore them in general algorithms. However [KvD89] investigated the geometry induced by highlights by describing their appearance under viewer motion. [Bla85] and [BB88] study the disparity of specular highlights in a stereo setup. [ZGB89] proves the existence of a 1-degree family of surface curves corresponding to a tracked highlight. [Nay88] uses two specular spheres to emulate a stereo setup requiring less calibration.

In the rest of this section, we will summarize the prior work that focused on the goal of the reconstruction of purely specular surfaces. Two broad classes of methods exist, the first one directly extracting geometric properties of the surface from observations, and the second one using some kind of integration around seed points. Finally, we will briefly introduce related work where specular surfaces are the main object of the study, but where the techniques used are far away from the computer vision domain.

1.2.1 Geometrical Approaches

Oren and Nayar [ON95b] investigate surface profile recovery from the tracking of unknown scene features seen through the reflection by a smooth curved surface. The setup requires a fully calibrated camera along with matching and tracking of reflected scene features. They show that in the 2-dimensional case, where camera motion, scene feature and corresponding virtual feature are coplanar, the 2D profile can be recovered uniquely from the tracking of 2 correspondences, with the constraint that the paths of the images of the virtual features overlap to some extent. The concept cannot easily be extended to the 3 dimensional case, as this requires some additional constraint which is obtained by using occluding contours, and as such is equivalent to having a strong prior on the surface itself.

An industrial application of specular surfaces in a very constrained scenario is given in [NSWS90], where Nayar *et al.* describe the implementation of a structured highlight

inspection system, applied to the quality control of soldered joints. It does not use virtual features, but highlights produced by an array of light emitting diodes. They assume distant light and observer, and recover surface orientation in the form of extended gaussian images (EGI), that can then be easily compared to reference ones to determine the acceptability of the measured joint.

Zheng *et al.* [ZM98] study the highlights produced by two circular light sources on a specular object placed on a turntable. In the absence of self-occlusion, this setup ensures that every surface point produces two specular highlights whose locations are used to extract surface position and orientation. Unfortunately, the method lacks the preciseness required and only produces rough models.

In a stereoscopic setup, Knauer *et al.* [KLKH05] use a similar constraint as in chapter 2 in order to perform, at each pixel, a one dimensional minimization on the depth estimate of a specular surface.

Finally, Savarese *et al.* [SP02],[SCP04] detail the mathematical constraints that arise when viewing the reflection of a calibrated pattern of intersecting lines in a specular surface that is represented by its special Monge form [CG00]. Although very elegant, their method requires image measurements such as local curvature and scale, and as such only allows the reconstruction of a handful of points.

The methods we propose in chapters 2 and 3 both also belong to these geometric approaches.

1.2.2 Integration-based Methods

When relatively dense correspondences between image features and scene features can be obtained, an often used technique is to integrate around seed points, using parametric surface models, or just points and orientations. For a given depth estimate, the corresponding surface normal is easily obtained, therefore giving an estimate of the depth of neighboring points, when the surface is supposed smooth. The advantage of such methods is that usually only a single image is required for the reconstruction. Inconveniences include the need of a-priori initialization, and a tendency of error accumulation over the iterations.

A characteristic implementation of integration methods can be found in [Sch96], in a method applied to the reconstruction of the surface of the ocean. Assuming distant illumination and a known radiance map of the illumination, camera pixels are matched with their incoming direction by intensity, and points on the surface are reconstructed by propagating around initial seed points without further assumptions.

In [PC96], Park *et al.* apply a similar principle but with the use of a laser beam coupled with a retro-reflective surrounding half dome, for the measurement of smooth specular surfaces. Starting from a known seed point, the orientation of the laser beam

and the position on the dome where the reflected beam is retro-reflected, they determine surrounding surface positions and orientations. The major inconveniences are the need for initialization, and more importantly the fact that the approximations on the surface geometry are accumulated over the iterations, which result in very imprecise measures far away from the seed point.

In [HBKM96], Halstead *et al.* detail a device used for the sub-micronic reconstruction of the human cornea, with the ambition of detecting irregularities to be clinically treated. Starting from a known seed point, they iteratively integrate along the computed surface normals to fit a spline surface to the cornea's true shape. The high level of accuracy they obtain is due to the highly constrained scenario in which they operate, where a spline surface closely resembles the target surface. The initialization they require is obtained by other means, but is clearly capital to the method's correctness. A similar method is proposed by [SNG03] to assist in the design of generic catadioptric mirrors.

In the same spirit, Tarini *et al.* [TLGS03] obtain the model of a smooth specular surface from the dense matching of image points with scene points on a target plane. The family of possible surfaces leading to the same point correspondences forms a 1 degree of freedom family, dependent on the initial depth chosen for the starting seed point. They argue that minimizing the *auto-coherence* of the recovered surface with respect to this initial depth converges to the true initial depth. The auto-coherence of a set of points and normals that should belong to a smooth surface represents the fact that neighboring points should have surface orientations that are coherent with one another. Given an initial point on the surface, different depths for another non-neighboring point are obtained by integrating along different possible paths between the two points. The auto-coherence is then quantified by the variance of the depth of the second point.

Solem *et al.* [SH04] treat the problem in a variational framework, in the very constrained scenario of a smooth surface and known contour. Their method applies for extending known structure to incorporate specular parts, for example a car window extended from the model of the bodywork. Supposing known correspondences, they fit a spline surface that minimizes the integral of the gaussian curvature over the surface. In [SsH04] they formalize the problem in a level set framework.

In a rather different setup, a reconstruction technique uses images acquired with polarization filters. Effectively, when an unpolarized ray of light is reflected by a specular surface, it becomes partially polarized in a direction orthogonal to the plane formed by the incident ray and the surface normal at the point of reflection. The phase image that is acquired through a polarization filter therefore encodes this plane of reflection, from which the true surface normal must be extracted. Rahmann *et al.* [RC01] use a second polarization image, therefore uniquely determining surface orientation as the intersection of two planes. A shape from shading technique is then used to recover the surface location for a given orientation.

1.2.3 A Word on Human Perception

The perception and interpretation by the human visual system of shape through specular reflection only (*i.e.* without additional cues such as contours or matte shading) is an open subject. In a founding paper, Blake *et al.* [BB90] show that the human visual system can interpret the depth information carried by specular highlights. In [SFFP04], Savarese *et al.* conclude that the human visual system is only slightly better than chance at finding the shape of mirror surface, when using the reflection of a regular grid. Flemming *et al.* in [FTA04] arrive to exactly the opposite conclusion, when using the reflection of real world environments instead of regular grids. Norman *et al.* [NTO04] also notice that the most favorable case for shape recognition is in the presence of specular highlights. They argue that the compression of the apparent texture that occurs in areas of high curvature is an accurate source of information concerning surface orientation.

1.2.4 Metrology

The inspection of specular surfaces is a common task in industrial environments. The most common example is found in the automotive industry, for the quality control of the bodyworks and windshields, but other *niche* applications could include the inspection of reflectors in optical instrumentation, or the finish in jewelry. Traditionally, this inspection is accomplished using contact probes, which is a time consuming task that is not applicable to mass production. Therefore, optical based techniques are beginning to emerge, with the objective of rendering the process more automated and rapid, enough not to slow down the troughput of the manufacturing production lines.

Laser Scanners

Obtaining 3D models of surfaces, sometimes also called reverse-engineering, with a high level of accuracy and a quantifiable error is traditionally done through the use of a laser scanner. These systems allow a quick and precise measurement, and are available for large scale applications such as architectural measurements, to much smaller ones, such as the inspection of small manufactured objects. Whereas the prices and availability of such systems tend to make them affordable for businesses and institutions, they are limited to opaque surfaces, as at least part of the energy released by the laser must be captured by the associated captor, which is in a fixed position. This is not the case for surfaces where the Lambertian component of the BRDF is negligible, such as mirror surfaces, because all the energy is reflected in the mirror direction, instead of leaving a characteristic *dot* on the surface. For these types of surfaces, the acquisition of a model with a laser scanner is only possible by artificially adding a Lambertian component to

the surface, usually by painting it or sprinkling it with chalk. Obviously, this solution is not very flexible, and even impossible for certain applications.

Recently, hybrid laser scanners have been constructed and commercialized for the specific application of specular surface inspection [BOIK01]: several parallel plates are attached in front of the CCD captor ensuring that the observed incident light arrives from a known unique direction. The major disadvantage is that an acquisition is much more time consuming than for traditional surfaces, as the captor must be swept through multiple positions for a each point. Furthermore, the system requires that the captor be at known positions at each instant, adding a possible source of error to the measurement.

Deflectometry

Another system, more anecdotal in the field of this thesis, but interesting because specifically tailored for reflective surfaces, is called phase measuring deflectometry [KKH04]. The results obtained are not a 3D model of the specular surface, but the derivative of a curvature map as seen from a single viewpoint. As such the method is used for the inspection of the micro-structure of the finish of a surface, as irregularities of the order of the micron can be detected.

2.

Volumetric Space Carving Approach

This chapter presents a specular surface reconstruction technique in the spririt of the space carving framework of Seitz and Dyer [SD97] and Kutulakos and Seitz [KS00]. The original framework supposes photometric invariance of a given surface patch with respect to observation location, a property which is not respected by specular surfaces. The method we propose relies on *geometric* invariance, a property which is verified whenever the object is static throughout the different images we have of it. Supposing the same constraints as the original space carving framework, plus matches between camera pixels and their source scene points, we show how surface orientation can serve as a consistency cue.
Work published in [BS03]

2.1 Space Carving Algorithms

A considerable amount of work has been carried out in the computer vision community over the past years in a reconstruction technique referred to as space carving. In the spirit of these space carving approaches, we present a novel consistency measure that extends the field of applicable surfaces to incorporate purely specular objects.

In a first section, we will recall space carving reconstruction techniques in the general case, and then present our extension to purely specular surfaces, along with results on real objects. We deliberately omit all the aspects inherent to visibility and self occlusion, as in our case they are an implementation issue, and can be incorporated straightforwardly.

2.1.1 Comparison with traditional geometric approaches

In some sense, space carving algorithms work in reverse order compared to more traditional geometric reconstruction techniques. Figure 2.1 illustrates a traditional surface reconstruction framework, where image primitives are first identified and matched in the input images, and different geometric constraints are then applied to these extracted primitives so as to compute geometric properties of the corresponding surface element. For example, in a stereo-vision setup, pixels onto which an identical surface feature is projected are extracted, and the location of this feature is computed by intersecting the 3D lines formed by the camera projection centers and the corresponding image pixels.

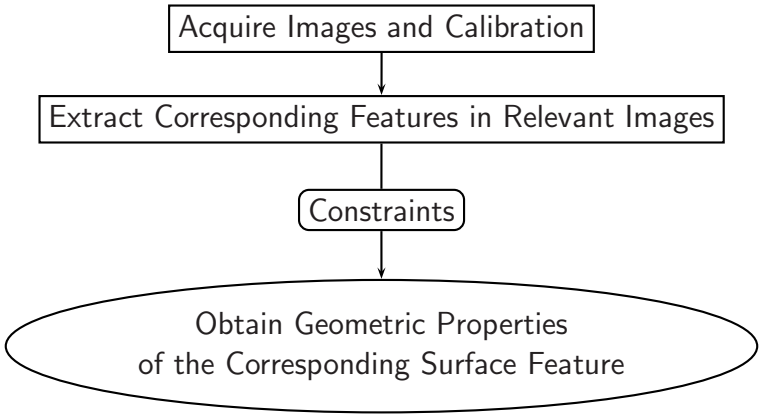


Figure 2.1: **Workflow of a typical surface reconstruction technique**

On the other hand, space carving techniques tackle the problem from the opposite direction, as illustrated in figure 2.2. Rather than directly identifying surface features in the images, and using some kind of constraint to determine geometric properties, space carving algorithms work by elimination, sweeping possible surface configurations

and verifying if a given configuration is compatible with the information available at hand, *i.e.* the corresponding pixel values.

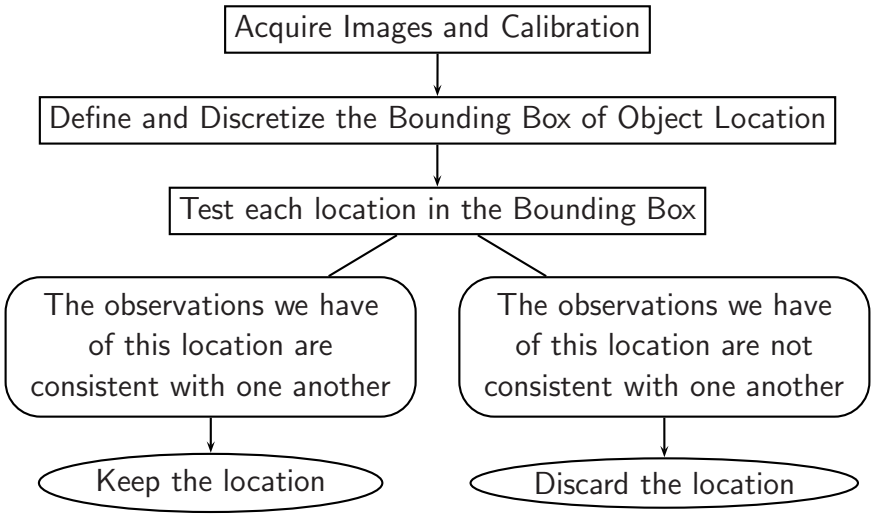


Figure 2.2: **Workflow of a space carving reconstruction technique**

2.1.2 Summary

As summarized in table 2.1.2, space carving techniques have a different field of application than traditional stereo methods. Their flexibility and generality allow artificial and natural scenes to be easily reconstructed, but their lack of precision in real situations forbids its use where such accuracy is primordial.

Algorithm 1 Space carving algorithm. (simplified version, without visibility handling)

```
1: aquire images from various locations
2: for each camera do
3:    compute internal calibration
4:    compute pose estimation
5:    eventually compute photometric calibration
6: end for
7: Determine bounding box and sampling of the object location
8: for each voxel in bounding box do
9:    for each camera seeing the voxel do
10:      project the voxel in the camera image plane
11:      determine the color of the corresponding pixel(s)
12:   end for
13:   Keep or discard the voxel depending on the different colors we have of it
14: end for
```

Requirements

- **Camera Calibration** There are no supposed constraints concerning the placement or the internal properties of the cameras used for the acquisition of the object images. The only requirement is that there must be a known projection function between a point in 3D space and its corresponding pixels in the input images. As such, camera calibration (internal and external) is supposed to be known, and must usually be performed with traditional image based calibration techniques (that usually require feature matching), or must be fully known *a priori*, for example by using a robotic manipulator, or a precalibrated environment.

  This calibration constraint is in fact very limiting towards obtaining a flexible reconstruction method, as all the pose parameters of the cameras must be expressed in a unique global coordinate system. Obtaining such a global calibration does impose some positioning constraints, for example by requiring certain features to be visible by several cameras. Chapter 4 will give insights as to more flexible techniques for obtaining this calibration.

- **Illumination** For the moment, we have presented the space carving framework as a technique that discards object locations that are not *photo-consistent* with one another. This very general denomination actually stands for a comparison of pixel intensities, as this is the primary data that is given by the pixel values of the different input images. The theoretical framework supposes that given these observed intensities (along with viewpoint locations), we have a mean of deciding

if they correspond to a sampling of a possible surface reflectance. In practice, a given 3D location is discarded or kept depending on the variance of the color of the corresponding pixels. This in practice imposes that the pixel intensities of a surface point be constant over the input images, which in turn implies that:

- The illumination of the object must be constant throughout the acquisition. This constraint is obvious, as it is clear that changing the color, intensity or incoming direction of the illumination of a point will change the appearance of that point whatever location it is viewed from.
- Given fixed illumination conditions, the appearance of a surface point must not depend on where this point is viewed from. This implies in most cases that specularities must be discarded or ignored, either manually or by imposing constraints on the acquisition so that these specularities are minimized. [YPW03] relaxes this constraint in the simplified case where imaged color is supposed to consist of an intrinsic color (the color of the surface) added to a varying amount of illuminant color. A voxel then satisfies the photo-consistency test if its associated colors form a line passing through the illuminant color.
- The cameras used for the acquisition should also be color calibrated, which is to say that a ray of light of given intensity and wavelength must be mapped to the same pixel values by the different sensors. In practice this imposes that their response function in each color band must be known beforehand, or at least accounted for, or that all the images be taken by the same acquisition device.

Accuracy Concerning the results that are obtained, space carving methods are limited to the configurations that have been tested. Given the discrete nature of these possible configurations, such algorithms have a trade-off between the accuracy of the reconstructed model, and the execution time necessary to test all possible configurations. In practice, the programmer must discretize the volume around the object in finite volume elements called *voxels*. The accuracy of the reconstructed model is directly related to the size chosen for these voxels, and the execution time is directly related to their number. As such, dense voxelic discretizations will produce sharp results, but at the cost of a high execution time and memory consumption, and inversely, coarse discretizations will produce results requiring low execution times, but at the cost of an imprecise reconstruction.

On the other hand, traditional constraint methods produce results that are not discrete (up to computer precision, which we will here assume as infinite). As the reconstructed model is obtained from the numerical solutions of the equations that define

the different constraints, the possible configurations are neither discrete, nor limited in accuracy.

Initialization In order to compute geometric constraints, traditional reconstruction algorithms suppose that surface features have been matched in the input images. Given the image of a particular point in one input image, some technique has to be used to compare pixel intensities in the other images to determine which pixel corresponds to the same surface feature.

The easiest way to obtain correct matches is to locate salient features, and manually identify these features in all input images. In case the input images are calibrated, epipolar geometry can be applied to reduce the possible matching locations to a specific line in the image, rather than search the entire image. Usually only a few surface points are sufficiently salient to be robustly extracted and matched in all input images, which in turn induces that only a few points of the surface can be easily reconstructed.

Space carving techniques do not require these matches, as they work directly with the pixel values of the input images, and therefore are better suited for unsupervised reconstructions. On the other hand, another kind of initialization has to be given: as these algorithms sweep possible object locations, some bounds to these possible locations have to be determined, of which the automation is a non trivial problem.

	Space Carving	**Traditional**
Obtained Results	- Accuracy limited by sampling the reconstruction volume - Location of reconstructed features fixed inside user defined bounding box	+ Accurate geometric properties + No limit in object location
Initialization	+ Only pixel information in input images - Need to specify a bounding-box of object location	- Supplied matches of corresponding object features

Table 2.1: Summary of advantages and inconveniences of space carving methods compared to typical geometric ones.

Implications The constraints that have emerged in the previous paragraphs clearly prohibit the use of space carving with specular surfaces, as the reflection of the sur-

rounding scene on such a surface clearly violates the photo consistency measure used to discern real surface features from empty space. In the following sections we will present an extension to the space carving framework that allows models to be reconstructed from the reflection of a known environment, and that is based on the geometric consistency of a surface feature rather than its photometric consistency.

2.2 Application to Specular Surfaces

Can the space carving framework be applied to specular surfaces, and if so, at what cost?

Clearly photoconsistency at a given object location is not an option, as specular surfaces violate the basic principle that an object's appearance is independent of where it is viewed from: the intensity of this location recorded in the different images depends on the object's environment and the point of view, rather than on the object itself.

What we can however assert is that the structure of a rigid object is independent of where it was viewed from. More specifically a given point on the surface has a specific orientation and position, and these properties are intrinsic to the object, not to the conditions of its observation. The method we present in this chapter derives from this simple fact.

Substituting geometric consistency to photometric consistency has a strong implication on the ease of use of the space carving approach: whereas obtaining pixel intensities from the input images is straightforward, given the images, obtaining geometrical information for a given pixel brings us back to the problem of feature matching, along with its inherent complexity.

The primary advantage that remains is the generality of the approach: the problem of specifically matching features throughout all the input images is a tedious and error prone task even in the general case, and becomes very arduous when these features are seen through the reflection by a specular surface, as they are deformed according to the geometry of that surface. By using a space carving approach, we do not require that a match be available in every input image, but instead use only the matches we are *able* to determine. As such the algorithm treats all available data, without having to explicitly account for the samples where only a subset of all measures could be used, due for example to the impossibility of determining a feature match in a very deformed area.

2.2.1 Geometric Constraints

Let us suppose the setup of a typical space carving framework, except that the object to reconstruct is a mirror rather than Lambertian surface.

Requirements

Applying the space carving framework to specular surfaces requires different constraints to those in the Lambertian case. For a given point in space to test, we will be computing the consistency of its geometric properties rather than its photometric ones. This means that we must have some way of computing these geometric properties, which in practice brings us back to some of the constraints of traditional geometric algorithms, namely feature matching, this time between scene features and their reflection in the images, rather than between images. On the other hand, the constant illumination constraint is of course lifted, along with the photometric calibration. In addition to the requirement of internally and externally calibrated cameras, we therefore also suppose having a means of knowing the 3D coordinates of the scene points reflected by the specular surface onto the camera image planes.

We summarize the requirements and setup:

- A set of fully calibrated cameras (internal calibration and pose), looking at the specular surface.
- Calibrated scene points around the specular surface.
- Matching between camera pixels and their source scene points.

Information given by one view

Let us consider a voxel in the chosen bounding box around the specular surface to reconstruct. As the cameras are completely calibrated, we are able to project this voxel onto the a point of the image plane of any input camera, by using this camera's projection matrix. Considering one of the input cameras, the pixel this voxel projects onto can be of three natures:

- the pixel does not belong to the image plane, *i.e.* the current voxel is not seen by the camera
- the pixel belongs to the image plane, but we were not able to find a match between this pixel and its origin in the scene around the specular surface.
- the pixel belongs to the image plane and we were able to compute the 3D location of the point that was reflected by the specular surface onto this pixel.

Of course, the first two situations do not allow any geometric property to be determined, and in this case we ignore the contribution of the given camera. On the other hand, the last case allows us to compute the orientation, *i.e.* the normal, of the specular surface at the given voxel, as shown in figure 2.3, as the bisector of the ray going from the voxel to

the 3D scene point, and the ray going from the voxel to the camera's projection center.

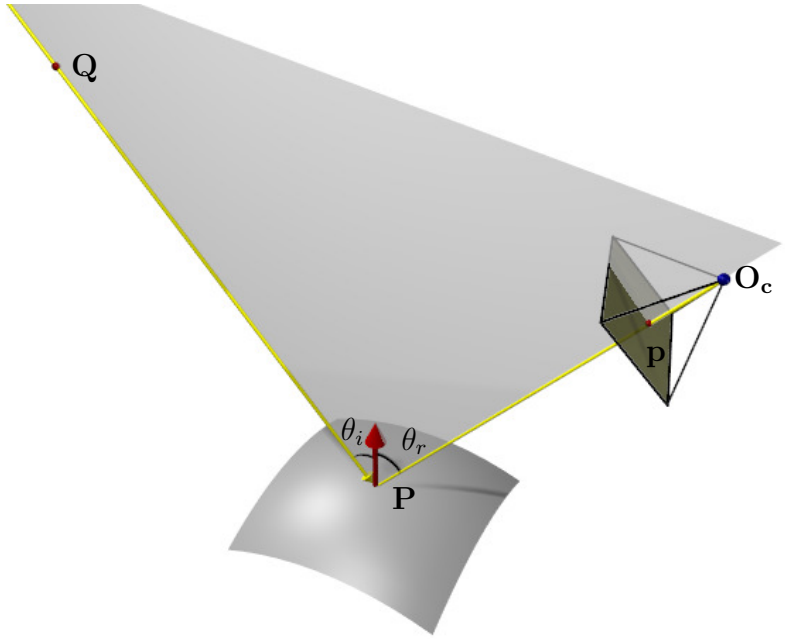


Figure 2.3: **Computing surface orientation.** $\mathbf{P}$ is the voxel currently being tested. The given match $(\mathbf{p}, \mathbf{Q})$ can only be produced if the normal at $\mathbf{P}$ is the bisector of the rays formed by $\mathbf{P}$ and $\mathbf{O_c}$ on the one hand, and $\mathbf{P}$ and $\mathbf{Q}$ on the other. The orientation of the surface associated to the voxel is therefore this bisector.

If we focus our attention camera-wise instead of voxel-wise, the information given by a match, represented in figure 2.4, is the discretization of the 1-dimensional family of possible surface configurations.

Multiple views

Let us suppose that the voxelic discretization of the bounding box around the specular surface is infinitely precise, and that there is no noise either in the projection functions of the cameras, or the matching of the camera pixels with their corresponding scene points. It is clear that for a voxel that belongs to the specular surface, all the normals that will be accumulated for all the input cameras will be identical to each other, as they will all correspond to the true orientation of the specular surface at that voxel. Figure 2.5 illustrates this property.

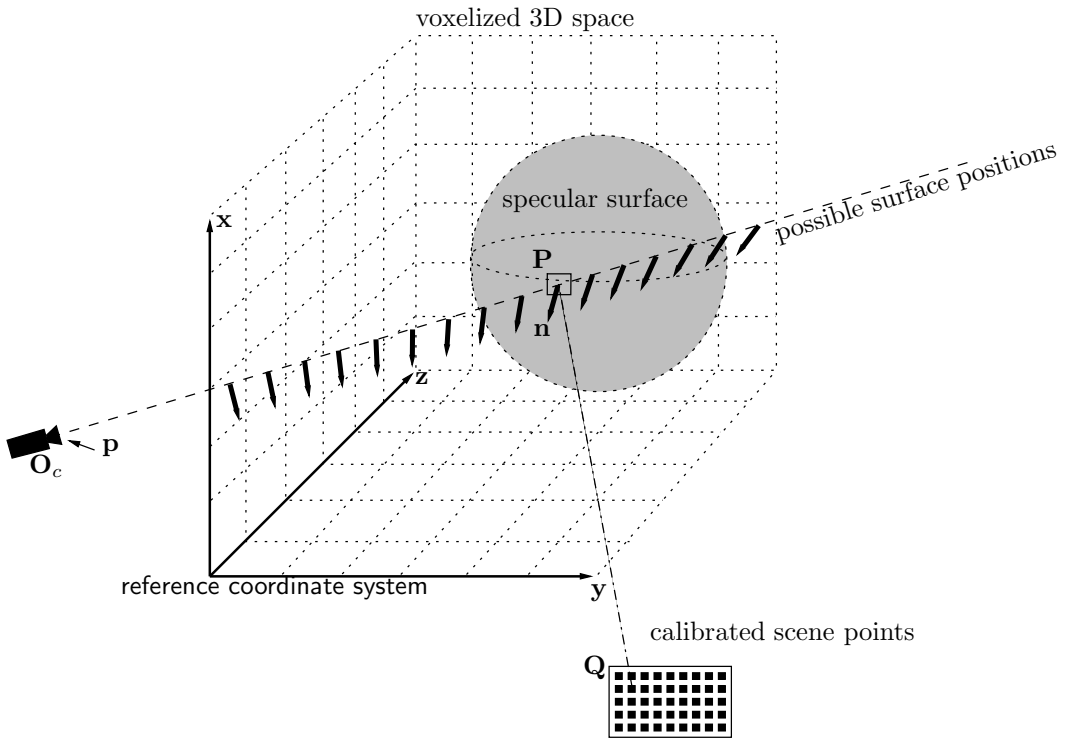


Figure 2.4: **Required Setup.** We here represent one of the input cameras. Given a match between a scene point $\mathbf{Q}$ and a pixel $\mathbf{p}$ of this camera, the information obtained is the one dimensional family of possiblesurface posititions with associated surface orientations, here discretized in a voxelic environment.

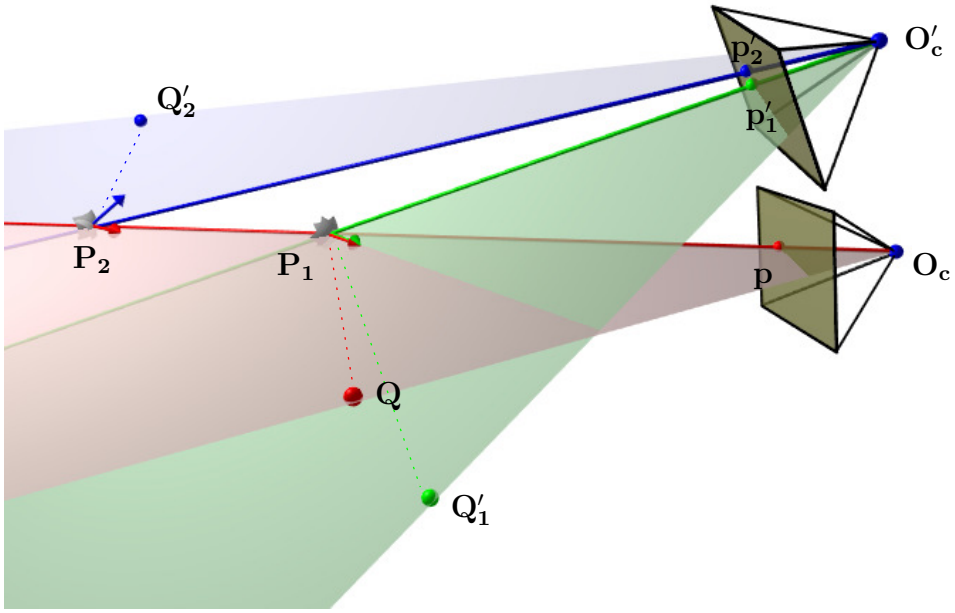


Figure 2.5: **The information available from multiple matches at a specular point.** Two possible surface positions $\mathbf{P_1}$ and $\mathbf{P_2}$, and orientations are represented along the red projection ray, given the match between pixel $\mathbf{p}$ and point $\mathbf{Q}$. Given a second view of these possible surface positions, we obtain the following information: The blue normal calculated at $\mathbf{P_2}$ given the match between $\mathbf{p'_2}$ and $\mathbf{Q'_2}$ in the second camera is inconsistent to the normal obtained with the first camera. The green normal calculated at $\mathbf{P_1}$ given the match between $\mathbf{p'_1}$ and $\mathbf{Q'_1}$ in the second camera is identical to the normal obtained with the first camera. As such, the only possible solution for the depth and orientation of the specular surface at pixel $\mathbf{p}$ is given by $\mathbf{P_1}$. Note that to ease representation the scene points $\mathbf{Q}$, $\mathbf{Q'_1}$ and $\mathbf{Q'_2}$ are located in the field of view of the cameras, but that this need not be the case.

We have therefore derived a consistency test for a given voxel to test: a voxel is discarded if the normals we have accumulated for it differ from one another, or marked as belonging to the specular surface if all its normals are identical.

2.2.2 Consistency Measures

In practice however the voxelic discretization is finite, and the calibration of the cameras, the estimation of the scene point coordinates, and the matching of camera pixels with their corresponding scene point are all a source of noise and error. This leads to the fact that even for a voxel that intersects the specular surface, and as such should be marked as correct, the normals that we have accumulated will not be strictly identical, which means that some measure of normal consistency must be elaborated.

Surface Normal Variance

We employ a classical variance measure on the computed surface normals in order to quantify the disparity of the surface orientation at a given voxel. We first compute the mean surface normal $\overline{\mathbf{n}}$ as

$$\overline{\mathbf{n}} = \frac{\sum_{i=1}^{n} \mathbf{n_i}}{n}$$

with n the number of successfully computed normals for the given voxel. $\overline{\mathbf{n}}$ is then normalized. The quantified disparity measure is then computed as the variance of the angles between each computed surface normal $\mathbf{n_i}$ and the mean normal $\overline{\mathbf{n}}$. As the surface normals have a norm of 1, the angle between $\mathbf{n_i}$ and $\overline{\mathbf{n}}$ is computed as the inverse cosine of the dot product of $\mathbf{n_i}$ and $\overline{\mathbf{n}}$:

$$\widehat{(\mathbf{n_i}, \overline{\mathbf{n}})} = \arccos(\mathbf{n_i}.\overline{\mathbf{n}})$$

The variance of the normals at a voxel $\mathbf{P}$ is therefore finally calculated as

$$\sigma^2(\mathbf{P}) = \frac{\sum_{i=1}^{n} \widehat{(\mathbf{n_i}, \overline{\mathbf{n}})}^2}{n}$$

Heuristic Disparity Measures

Using only variance as a disparity measure is in practice insufficiently discriminant in the case where the cameras tend to be placed in the same direction with respect to the specular surface, as is usually the case due to matching and calibration constraints. As well as normal disparity, depth information must be taken into account, as also noted

by [TLGS03], and illustrated in figure 2.6: indeed, whatever the match, at infinity the corresponding surface normal is equal to the inverse direction of the line of sight.

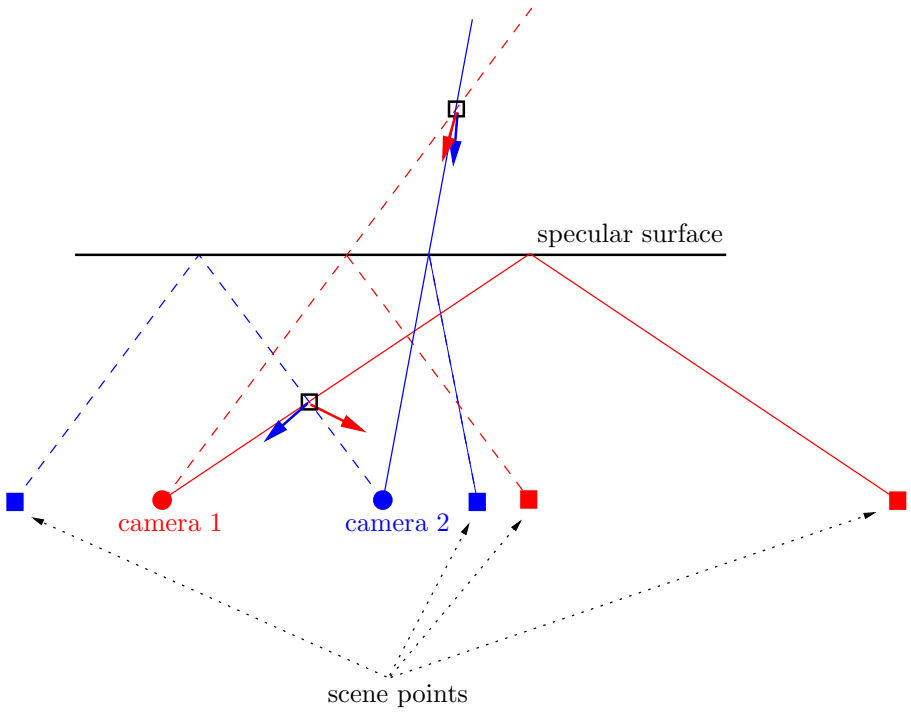


Figure 2.6: **Disparity variation due to camera distance**. Two equally incorrect voxels have an associated normal disparity that depends on their distance to the cameras. A threshold-based decision will tend to keep incorrect voxels if they are further away from the camera viewpoints.

This problem of calculated variance decreasing with distance could be overcome by a special reprojection error, when supposing that the scene points from which the matches were obtained originate from planar targets. For each voxel, given the mean surface orientation calculated by equation 2.2.2, the corresponding pixel could be back projected and reflected, thus computing its intersection with the corresponding target plane. The disparity of the given voxel would then be some function of the distance between each reprojected point, as shown on figure 2.7. We did not further investigate this solution, as it does not fit in the space carving framework.

Instead, we implemented a disparity measure that heuristically privileges voxels as they get closer to the camera by dividing the disparity obtained in (2.2.2) by the mean angle formed by the scene points, the voxel, and the different camera viewpoints. As the distance from the voxel to the cameras increases this mean angle tends to decrease,

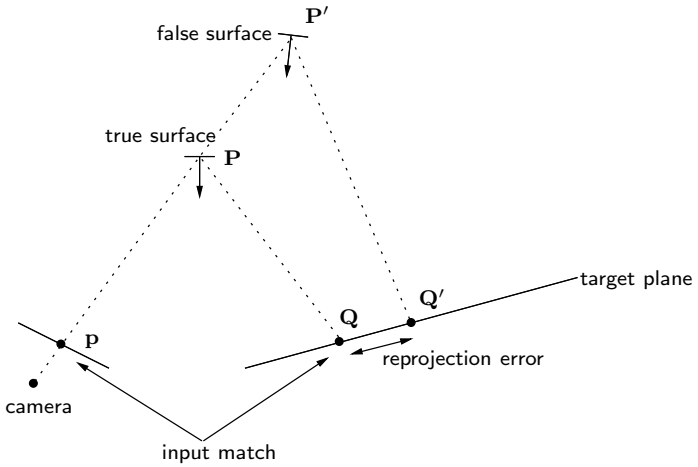


Figure 2.7: **Reprojection error.** Given an input match between $\mathbf{p}$ and $\mathbf{Q}$ obtained through the reflection at point $\mathbf{P}$, a point $\mathbf{Q}'$ on the target plane is obtained by "back-reflecting" $\mathbf{p}$ through $\mathbf{P}'$. The reprojection error for the voxel at $\mathbf{P}'$ can then be obtained by the distance between $\mathbf{Q}$ and $\mathbf{Q}'$.

thus favoring the disparity measure for closer voxels.

$$disparity(\mathbf{P}) = \frac{n\sigma^2(\mathbf{P})}{\sum_{i=1}^{n} \widehat{\mathbf{O_{ci}}, \mathbf{Q_i}}}$$

Extracting the final voxels

Calculating the disparity measure for each voxel of the discretization around the specular object results in a probability density function, from which we can extract a probable specular surface by applying a threshold. Choosing a value for this threshold greatly influences the quality and quantity of the extracted voxels, and is a task that is difficult to automatize throughout different setups.

An answer to this problem, which was also proposed in [YPW03], is to traverse ([AW87]) the voxels belonging to a projection ray, and keep the voxel which has the minimal surface orientation disparity. Of course, we only select a voxel if it satisfies minimal quality constraints, namely if its variance is obtained from a sufficient number of matches, and is under a specified tolerance. This optimization ensures that every match produces one and only one voxel in the final reconstruction (provided minimal quality is attained).

2.3 Experiments and Results

We validated the proposed method using two different approaches. Firstly, using computer generated raytraced images, we show results on the reconstruction of a specular sphere.We do not apply the optimization of the previous paragraph, in order to give a view of the normal disparity over the whole bounding volume. Secondly, we apply the method on a real reconstruction of a shiny spoon obtained from images of an off-the-shelf camera. For both of these experimentations, we only reconstruct a patch of the entire surface, due to the fact that the cameras and scene points are located on one "side" of the object.

2.3.1 Computer generated images

The results shown on figures 2.9 to 2.13 were obtained using 40 ray-traced [Pov96] images of a specular sphere. A black on white checkerboard target was reflected by the sphere onto the image plane of the cameras (see figure 2.8), and a Harris corner detector [HS88] was used to extract the corners in the obtained images, thus creating some noise in the measures. The extracted points were then analytically matched with their corresponding scene points on the target, using the known geometry of the sphere.

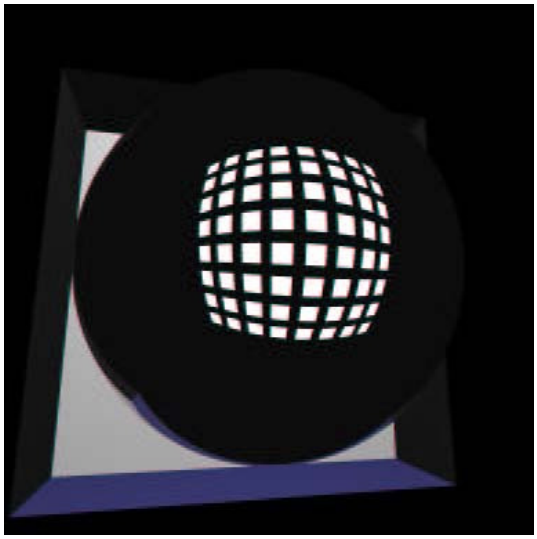


Figure 2.8: **Raytraced sphere** One of the 40 computer generated images used for the reconstruction of a specular sphere.

We have tested our reconstruction method with different thresholds, as shown in

table 2.2, using the heuristic disparity measure seen above. The quality of the output depends on the minimum number of normals a voxel must be associated with before the disparity measure can be trusted. With less than 4 normals per voxel, a number of outliers can appear. With at least 8 normals, the extracted voxels are never further than two voxel from the original surface.

	strict	medium	laxist
4	363 93.4%	969 95.3%	3103 94.45%
8	161 100%	589 100%	1982 100%

Table 2.2: Results obtained using strict, medium and laxist thresholds on normal disparity. The rows represent the minimum number of normals associated to a voxel before this voxel can be considered. The columns represent the threshold that was applied to normal disparity. For each combination, the first number is the number of accepted voxels, and the second one is the percentage of these voxels that are either intersecting the specular surface or less than two voxels away from an intersecting voxel.

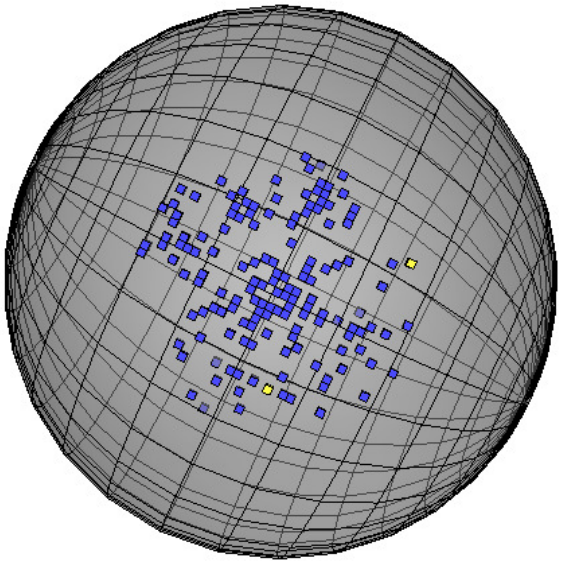


Figure 2.9: **synthetic reconstruction.** 161 voxels extracted on the sphere's surface using a strict tolerance on normal disparity. 158 dark (blue) colored voxels intersect the specular surface, while 3 light (yellow) colored ones are adjacent to an intersecting voxel. No further voxels have been extracted.

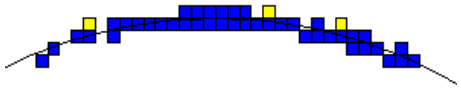


Figure 2.10: **synthetic reconstruction.** Cut through a reconstruction of 973 voxels.

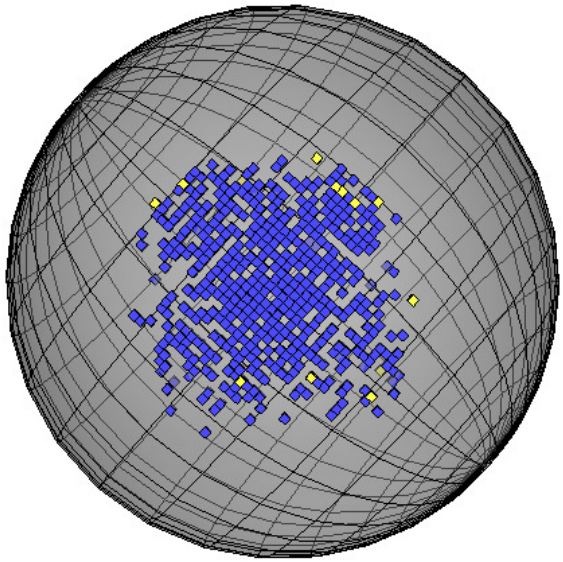


Figure 2.11: **synthetic reconstruction.** 589 voxels extracted on the sphere's surface, using a medium tolerance on normal disparity. 573 dark (blue) colored voxels intersect the specular surface, while 16 light (yellow) colored ones are adjacent to an intersecting voxel.No further voxels have been extracted.

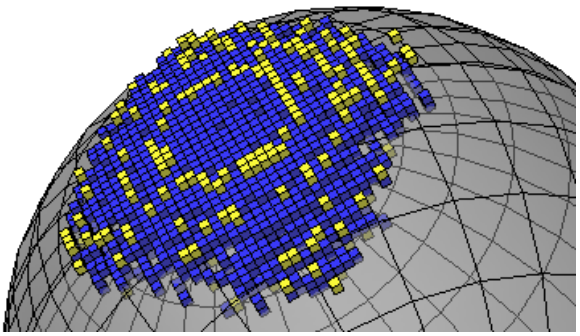


Figure 2.12: **synthetic reconstruction.** 1982 voxels extracted on the sphere's surface, using a broad tolerance on normal disparity. 1659 dark (blue) colored voxels intersect the specular surface, while 323 light (yellow) colored ones are adjacent to an intersecting voxel. Nearly all intersecting voxels were extracted, while no voxels that were further to the surface have appeared.

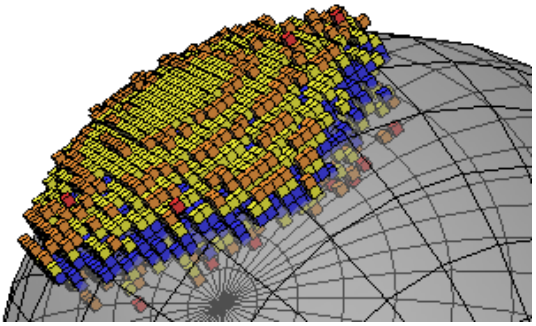


Figure 2.13: **synthetic reconstruction.** 4162 extracted voxels, using a very laxist threshold on normal disparity. 1934 dark (blue) colored voxels intersect the specular surface, while 1765 light colored ones are adjacent to an intersecting voxel. An additional 463 voxels (in orange and red) were more distant, however never being further than 5 voxels to the correct surface.

2.3.2 Real-world setup

We tested our method on the reconstruction of a part of a specular spoon made of low end polished metal, thus exhibiting consequent surface normal variation on some parts and substantial blurring (see figure 2.14, a little left from the center for example).

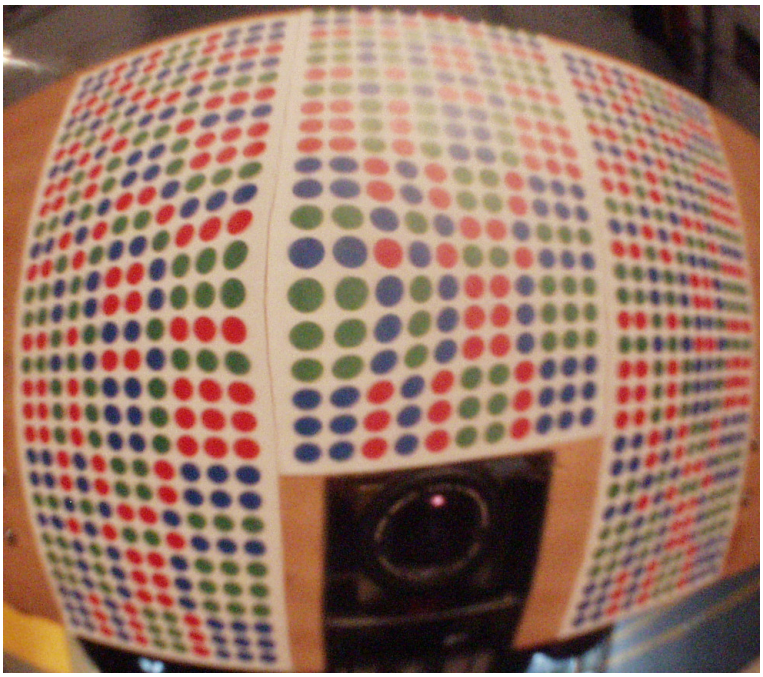


Figure 2.14: One of the 56 images used for reconstruction. Around 2 out of 3 colored circles were automatically extracted for this image.

Experimental Setup

Figure 2.15 shows the experimental setup. A color coded [MNY92] printed target was attached to an Olympus C2500 digital camera taking images of the specular spoon from 56 different viewpoints, while a stereo rig was used to obtain the pose of the target + camera system for every viewpoint. The unique color code allowed us to automatically match a pixel in the image corresponding to a circle, to the point on the target that was reflected onto this pixel, by using the color of the circle and of its 4 surrounding neighbours. The pose of the camera and the printed target was recovered with the method presented in chapter 4 page 100. We then extracted 1890 voxels corresponding to the specular surface out of a 100x100x100 cube of voxels, each of these voxels being 1 mm^3. The reconstructed patch spans about 10 square centimeters

of the spoon. The presented reconstruction is almost certainly incorrect to some extent, as the extracted scene points were not very precise, camera distortion was not modelled, and the computed poses were probably not perfect. Still, the results do show that the method is applicable in real setups.

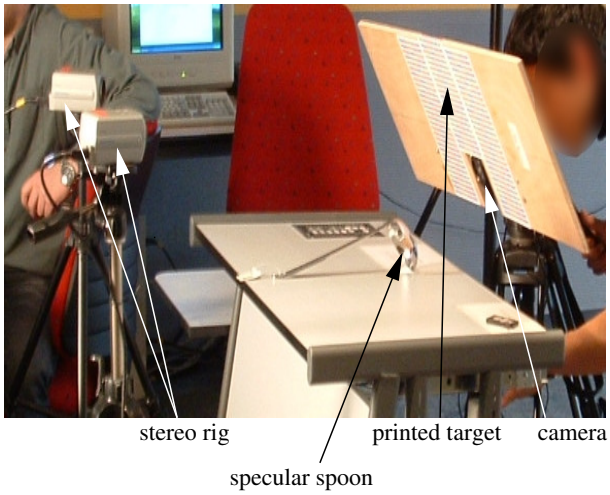


Figure 2.15: Experimental setup. The printed target is reflected by the specular sphere onto the image plane of the camera. The system's pose is determined by the stereo rig (see chapter 4 page 100).

Results

Reconstruction results for a patch of the specular spoon are shown in figures 2.16 and 2.17. What these images do not clearly show but can be noticed in the 3D model is that there are no holes greater than 1-2 voxels on the patch, and that the extracted thickness is around 1-3 voxels. The extracted voxels are qualitatively correct, and quantitatively correct as far as what we could measure.

2.4 Discussion

The method we have presented extends the space carving framework to incorporate objects with a sharp specular component[1]. It requires the same setup, with the additional constraint of having known correspondences between image pixels and corresponding

[1]This method has since been extended to refractive surfaces, namely the measurement of the surface of flowing water, in [MN05]

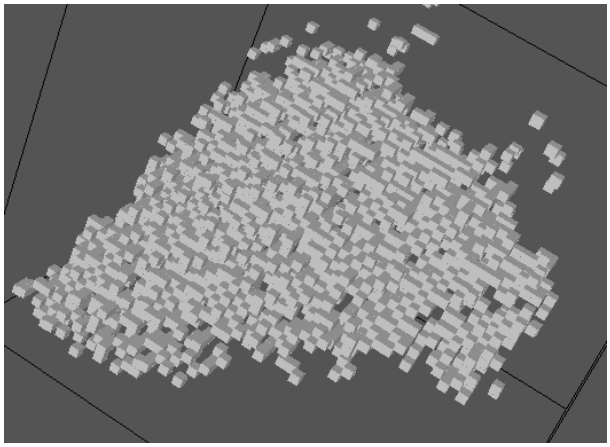


Figure 2.16: Front view of the reconstructed voxels.

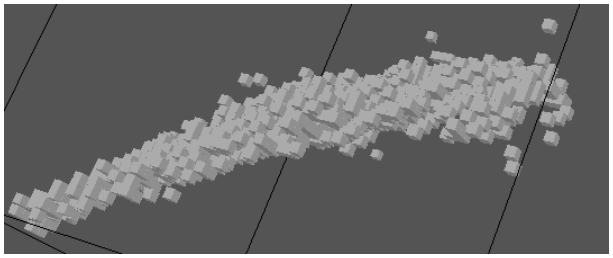


Figure 2.17: Side view of the reconstructed voxels. The spoon's curvature is clearly visible in one dimension. Views from other directions show curvature is correct on all the reconstructed surface.

calibrated scene points. Results on simulated data and experiments on real specular objects have shown that satisfactory reconstructions can be obtained. Compared to the state of the art, the primary advantage is that we do not assume surface smoothness[2], and therefore do not deviate from the true surface like integration based mathods tend to do. However the required setup does leave a rather bitter aftertaste: whatever the elegance of using geometric rather than photometric invariance, the method can be seen as having the inconveniences of both traditional and space carving approaches. The inconveniences of space carving approaches for matte surfaces, namely the need for initializing the bounding box and the lack of precision due to sampling are largely counterbalanced by the fact that no explicit matching has to be performed, which is not the case here.

[2]More precisely, the surface should be smooth but only at the scale of a voxel. Discontinuities and sharp angles when this is not the case will only cause the corresponding voxels to be eliminated, without offsetting the complete reconstruction.

This, coupled with the fact that camera calibration and pose must be known, limits the applicability of the method to very constrained scenarios, as this information cannot be trivially obtained due to the constraints imposed by the fields of view and focusing of the cameras. The consequence is we are left in a rather uncomfortable situation where the results produced are of insufficient quality for industrial or precise measurements, and the complexity of the setup prohibits the use of the method in real world applications.

The next chapter will present a triangulation method that relaxes some of these constraints, by limiting the number of input views to two, and using an immobile camera, thus requiring internal calibration but not pose.

3.

General Specular Surface Triangulation

This chapter presents a very simple method for obtaining a dense reconstruction of an arbitrary specular surface as viewed from a unique camera viewpoint. The surface is reconstructed point by point, relaxing the traditional constraints of surface continuity and/or smoothness. The geometrical constraints that allow this are obtained by viewing the reflection in the specular object of a planar target placed at two distinct but arbitrary locations. The method bares many ressemblances with the calibration of general (*i.e.* non-central) cameras.
Work published in [BSG06] (also independently published in [KS05])

3.1 Approach

We describe a method recovering points of a specular surface, independently from one another. We assume an internally calibrated pinhole camera viewing the reflection of a planar target, and a dense matching of the camera pixels with the points on the target. While the camera is rigidly attached to the specular surface, we acquire images of the reflection of the target placed at two different locations. The foundation of our method is closely related to the work on general (*i.e.* non central) cameras, as the reconstruction of the specular surface from the images of a calibrated camera is equivalent to the calibration of a non-central catadioptric system consisting of a central camera and an arbitrary mirror.

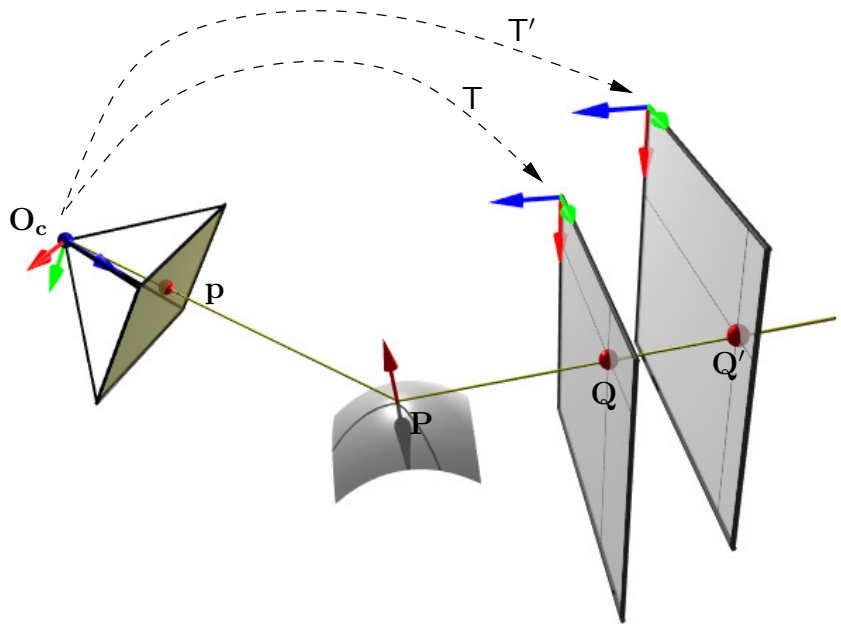


Figure 3.1: **Reconstruction Approach.** Matching of image pixels with their source in the targets and estimating two plane poses is sufficient to reconstruct the surface.

Suppose a calibrated pinhole camera located at $\mathbf{O_c} = \mathbf{0}_3$ observing the reflection in an unknown specular surface of two scene points $\mathbf{Q_i}$ and $\mathbf{Q'_i}$. This constraint is sufficient to uniquely determine the depth of the specular surface at $\mathbf{p_i}$, namely as the intersection of the lines formed by the camera's projection center and $\mathbf{p_i}$ on the one hand, and $\mathbf{Q'_i}$ and $\mathbf{Q'_i}$ on the other.

If we consider the (camera + specular surface) system as a general camera, finding two points $\mathbf{Q_i}$ and $\mathbf{Q'_i}$ for each $\mathbf{p_i}$, and therefore obtaining a reconstruction of the surface, is equivalent to calibrating this camera, as this is usually done as a one-to-one mapping of image pixels with lines in 3D space. In [GN01] or [SR04], such a calibration is achieved by using points on calibration planes: pixels in the image are matched with their 2D correspondent in the target planes, then the only step necessary in order to obtain 3D coordinates of these points is to estimate the pose of the planes in the camera reference coordinate system. Figure 3.1 summarizes our reconstruction method for 3 point correspondences: reconstructing the specular surface sums down to matching camera pixels with their source in the target planes, estimating the two euclidean transformation matrices T and T′ that map points from the target reference coordinate system to the camera one, and triangulating.

3.2 Discussion

The very simple method described here allows points of a specular surface to be reconstructed, independently from one another, therefore without having to assume continuity or smoothness. Actually, whereas the reconstruction in itself is straightforward, the requirements aren't as obvious. A method for matching pixels with their source in the target planes can be found in appendix A. Computing the poses of the target planes in the reference frame of the camera is also a practical problem, of which insights can be found in chapter 4.

The constraint we use for computing mirror point locations is actually independent of *specular* reflection. It is only based on the fact that light arriving at a surface point is reflected *at* that point, and not scattered. We could theoretically imagine any surface that locally modifies the direction of travel of light, as shown on figure 3.2, such as a surface with off-specular reflection, or more practically with one level of refraction, and obtain reconstructions by applying the method unmodified. Given the fact that in our case we *know* that we are treating specular reflection, the surface normal is at the same time calculated, as the bisector of the two intersecting lines.

Note that with real setups and thus with noise in the poses of the target planes and in the matching of the pixels, the lines formed by the projection center and the pixel on the one hand, and the two target points on the other will not exactly intersect. In this case the intersection will be defined as the median point of the shortest segment that connects the two lines.

We could also imagine using additional target planes: in this case, the second line should be defined as the best fit to all the matches.

In practice, we also perform a global non-linear optimization of the poses T and T′

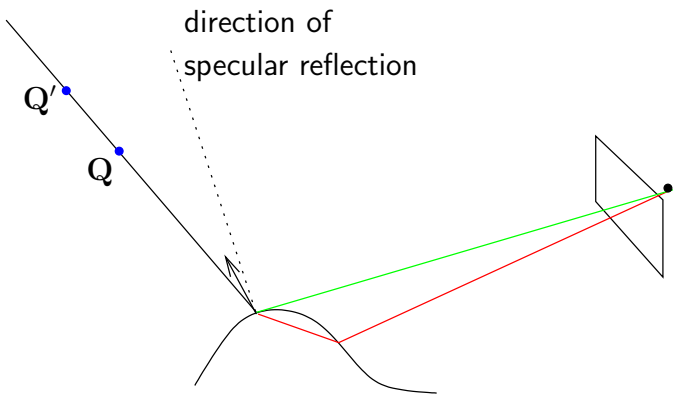


Figure 3.2: **Illustration of other possible and impossible configurations.** The green ray illustrates a theoretical configuration where the method could be applied unmodified, in the event of a surface with (very) off-specular reflection. In red, a non-applicable configuration for example if the object is transparent, as the light path consists of three rays instead of two (this case is formalized in [KS05]).

of the target planes, before the triangulation. The cost function to be minimized is the distance between matching lines in 3D space which we minimize using a Levenberg Marquardt algorithm:

$$cost(\mathsf{T}, \mathsf{T}') = \sum_{i \in \{matches\}} dist^2((\mathbf{O_c}, \mathbf{p_i}), (\mathsf{T}\mathbf{Q_i}, \mathsf{T}'\mathbf{Q'_i}))$$

3.3 Results

We present results when using the method in 3 different scenarios. The first setup consisted in the reconstruction of curved mirrors, the second of only perfectly planar mirrors, and the third of a car windshield, in an "out of lab" environment.

3.3.1 Results on curved mirrors

Our test setup (shown on figure 3.3) consisted of a 3 Megapixel digital camera mounted on a tripod. In the field of view of this camera we placed two wide angle rear view mirrors with a cheap coating, an ice-cream cup, and the platter of a computer hard drive. An LCD flat panel display was used to display structured light codes (see appendix A), and was placed in two distinct positions in such a manner that the patterns it displayed were viewed by the camera as reflected by our mirror objects.

Figure 3.3: **Experimental Setup.** Two of the images used for the reconstruction. Notice the 3 curved mirrors (an ice-cream cup and two small wide-angle rear-view mirrors), the planar hard drive platter, and a direct view of the target plane (the LCD panel), in the upper part of the image. The pattern displayed on the LCD panel corresponds to the structured light approach presented in appendix A.

Having no ground truth results, we evaluated the correctness of the method by fitting a plane to the part of the reconstruction we knew was planar, *i.e.* the hard drive platter (linear least squares fitting, without outlier removal). In the reconstruction shown on figure 3.4, over 98% of the computed points were less than 0.2 mm away from the surface, and 64% less than 0.1 mm. The approximate diameter of the reconstructed part of the platter was 80 mm, resulting in a maximum 0.3% relative error in the reconstruction. The histogram of these distances is shown in figure 3.5.

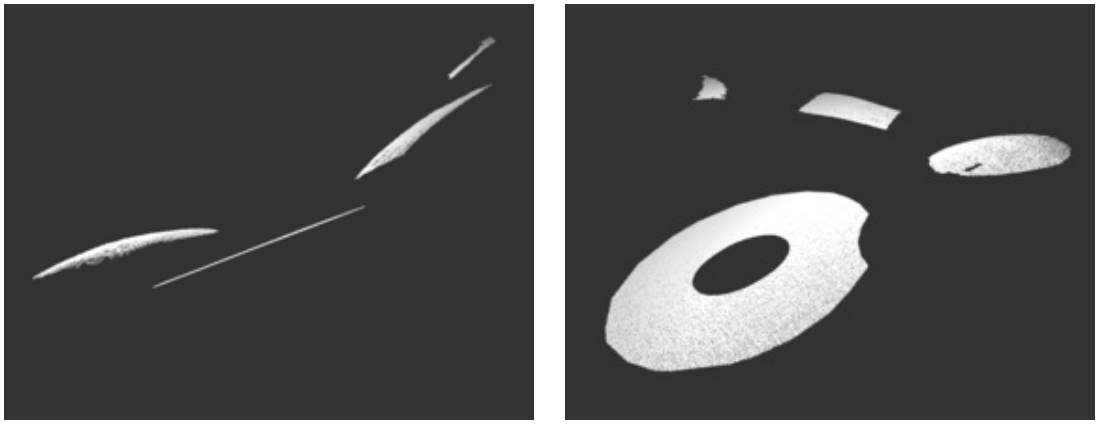


Figure 3.4: **Results.** Reconstructed surface viewed from two locations. The model contains over 525 000 independent points. Note the planarity of the reconstructed hard drive platter in the left image. Only a few points could be computed on the ice-cream cup, as its surface covered by the exploitable Gray codes was limited. The two small rear-view mirrors (one with circular, the other with rectangular based shape) were completely reconstructed (apart from a non specular dent in the circular one).

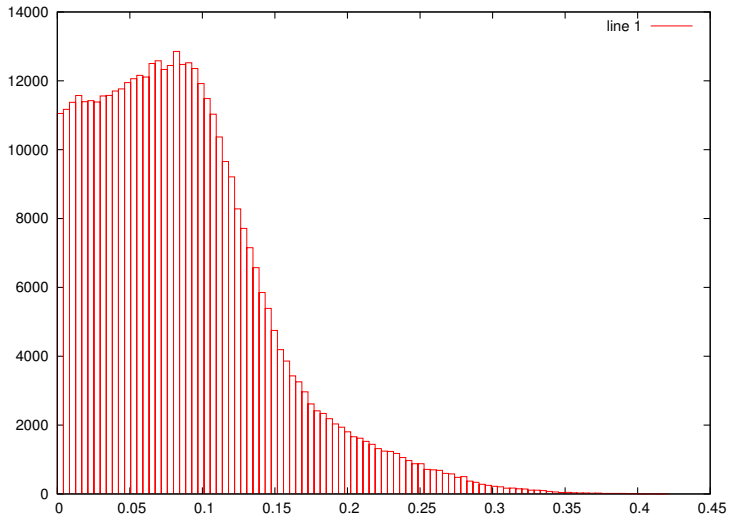


Figure 3.5: **Point-plane distance.** Histogram of the distance in millimeters of each point to the linear least squares fitted plane, after global optimization. The mean distance from the points to the plane is 0.086mm.

3.3.2 Results on perfectly planar mirrors

The accuracy of the reconstruction also depends on the quality of the pixel matching. Indeed, when experimenting with purely scenes containing only piecewise planar surfaces, as shown on figure 3.6, where the sub-pixel matching was more accurate because the structured light was not distorted, the distances to the fitted planes dropped down to 99.9% of the computed points less than 0.1 mm away from the surface, and 88% less than 0.05 mm. This is becaus the matches contribute to the optimization of the poses of the target plane, and thus to the overall quality of the surface reconstruction. The histogram of these distances can be seen in figure 3.7.

Figure 3.6: **Experimental setup.** This picture shows a view from our experimental setup. In the field of view of the camera, we see 4 planar mirrors, actually hard drive platters. At the top of the image, we have a direct view of the actual planar target. The curved mirror in the center of the image isn't used in this setup.

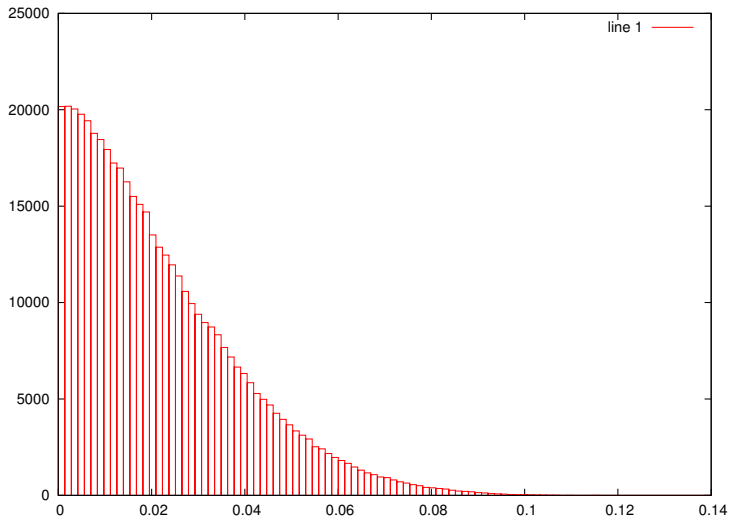


Figure 3.7: **Point-plane distance.** Histogram of the distance in millimeters of each point to the linear least squares fitted plane, without globally optimizing plane poses.The mean distance from the points to the planes is 0.026mm. In this setup the optimization did not significantly modify the poses of the target planes, producing seemingly identical results.

3.3.3 Results on a car windshield

The final experiment we present here was on a large scale object, namely a car windshield. We wish to thank the Techlab company for having let us use their equipment and provided us with the windshield for this experiment. The results presented here are mostly illustrative of the applicability of the method in an industrial environment, as it was here applied without modification. The setup, shown on figure 3.8, consisted of a tripod mounted camera, and a video-projector projecting the structured light on a board placed behind the windshield. Qualitatively, the reconstruction *seemed* correct, but quantitative results cannot be given here for the following reasons:

- The distortions produced by the projector were not modelled or taken into account.
- The patterns were projected on a non rigid board, that wasn't perfectly planar.
- The windshield itself isn't rigid, therefore its shape depends on its orientation and the contact points with its support. The CAD model we had of the windshield corresponded to another configuration.

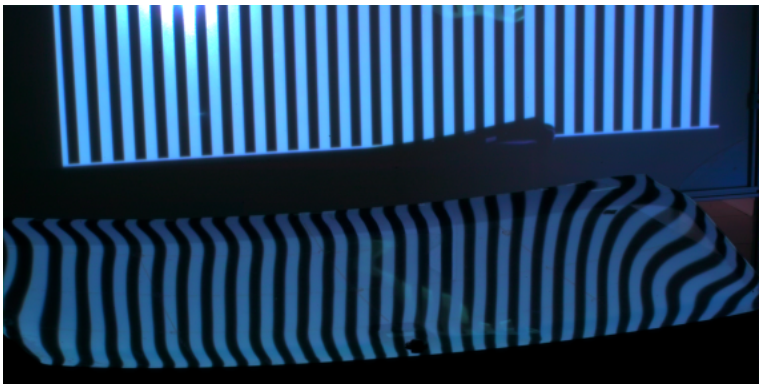


Figure 3.8: **Windshield reconstruction setup.** Reconstruction of a car windshield, which is placed such that we have a view of its concave side, in order to obtain the reconstruction of the whole part. The target plane we use here consists of a white board onto which we projected structured light from a video-projector.

3.4 Conclusion

We have presented a novel method that reconstructs a specular surface from two (or more) views. The proposed setup consists in a single camera, which is not necessarily

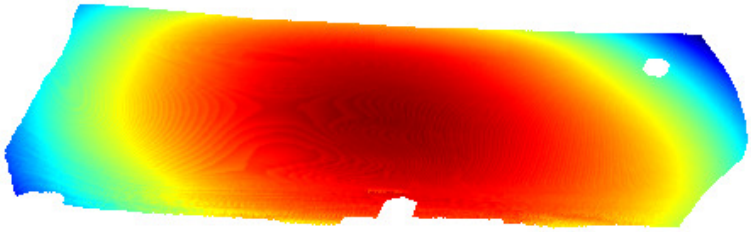


Figure 3.9: **Real World Reconstruction.** Reconstruction of a car windshield, the color encoding the distance to the best-fit plane. The method allowed us to easily obtain a 800 000 + point model using a classical video projector, on a large scale reflective surface. The hole in the middle is due to a non-specular patch on the surface.

a pinhole, viewing the reflection through the specular surface of known points from two calibration planes. A point on the specular surface is straightforwardly reconstructed by noting that the projection ray passing through the corresponding pixel, and the line formed by the two points on the target planes, must and can only intersect at the surface location. Compared to other reconstruction methods, we attain a high level of accuracy, without having the need to suppose surface continuity or regularity. We believe this method could easily be implemented in an industrial surface inspection application, at least to provide an accurate initialization for integration based reconstruction methods, probably the only purely vision based techniques able to detect surface micro-structure. Results showing the planarity of the reconstructed surfaces on areas we knew were planar were presented, along with the reconstruction of a large scale object in a real world setup.

Whereas the method in itself is obvious, the setup it supposes does pose practical problems. A point of the specular surface corresponding to a particular pixel can only be determined if the pixel was matched in *both* the target planes, thus supposing that we have a *dense* matching between pixels and target points, and not only extract a finite number of features from a printed target for example. A solution to this, using an LCD monitor and Gray codes is presented in appendix A, provides sufficient accuracy at the cost of having to display multiple images on the LCD monitor, thus requiring a rather long aquisition time. Furthermore, the poses of the target planes is supposed to be known. One can of course arrange for a setup where the target planes are in direct view of the camera, and then use classical plane based pose estimation techniques, but this implies that the camera does not focus *only* on the specular surface, thus reducing possible resolution. To this avail, the next chapter presents practical and theoretical methods for obtaining the pose of an object when it is not in direct view of a camera.

4.

Indirect Pose Estimation

In this chapter, we discuss the obtention of the pose of a camera with respect to a calibration object, in the case where the object isn't in the field of view of the camera. We first give relatively direct approaches, using either known mirrors or additional imaging devices, and then present a method using the reflection of the calibration object in 3 unknown planar mirrors. In this case we recover the pose of these 3 mirrors *and* the pose of the calibration object. Finally we present the theory for recovering the pose of two of such objects through the reflection of an arbitrary specular object, in a setup comparable to the previous chapter.
Work published in [BS03, BSG06, SB06]

4.1 Introduction

In this chapter, we discuss the obtention of the pose of a camera with respect to an object, in the case where the object isn't in the field of view of the camera. In the field of specular surface reconstruction from images, such situations are extremely common as:

1. The pose of the camera with respect to the reflected scene points it is viewing is a prerequisite for most if not all specular surface reconstruction algorithms.

2. The position of the camera with respect to the specular surface is strongly constrained by the fact that the reflection of the scene points must be seen on the part of the surface we wish to reconstruct. It can therefore be impractical or impossible to place a calibration object in a visible location.

3. In general the camera must be set to full zoom in order to obtain sufficient detail of the reflected scene points, decreasing the field of view.

All these parameters imply that it is often impractical and sometimes impossible to obtain a direct view of a calibration pattern at the same time as its reflection in the object to reconstruct. To overcome these limitations, we present some practical methods for obtaining the poses of such calibration patterns. We then investigate pose estimation using reflection of a calibration object in a mirror that we know is planar placed at different positions. Finally we introduce the geometric constraints that can be used for planar pose estimation through the reflection of an unknown specular surface.

4.2 Geometric Prerequisites

We will frequently be using the reflection of an object in a planar mirror. The geometry involved in these types of transformations is outlined here.

4.2.1 Reflections in planes

Consider a plane $\Pi = (\mathbf{n}^\mathsf{T}, d)^\mathsf{T}$ in 3D space, *i.e.* consisting of points satisfying the equation $n_1X + n_2Y + n_3Z + d = 0$. In the following, we will always suppose that the plane's normal vector $\mathbf{n}$ is of unit norm. The reflection in Π can be represented by the following transformation matrix:

$$\mathsf{X} = \begin{pmatrix} \mathrm{I} - 2\mathbf{n}\mathbf{n}^\mathsf{T} & -2d\mathbf{n} \\ \mathbf{0}^\mathsf{T} & 1 \end{pmatrix}$$

Let us denote the upper left 3×3 matrix of X by $\bar{\mathsf{X}}$. It is an orthogonal matrix, with determinant -1 (whereas a rotation matrix has determinant $+1$). Further, it has $+1$ as double eigenvalue and -1 as single eigenvalue. The plane normal $\mathbf{n}$ is an eigenvector of $\bar{\mathsf{X}}$ to the eigenvalue -1. Note also that as a double reflection is the identity, $\mathsf{XX} = \mathsf{I}$ and therefore $\mathsf{X}^{-1} = \mathsf{X}$.

4.2.2 Reflection in two planes

Consider successive reflections in two planes, as shown in figure 4.1. This is a fixed-axis rotation, with the intersection line of the two planes as rotation axis: the transformation preserves the intersection line of the two planes point-by-point, and thus is a fixed-axis rotation. Note that the rotation axis is orthogonal to the normals of both planes.

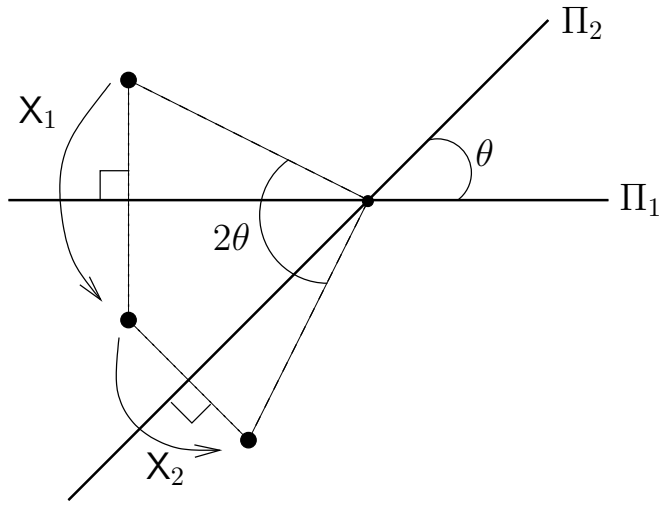


Figure 4.1: **Reflection in two planes.** Top view (along the axis of intersection of the two planes) of the reflection through planes Π_1 then Π_2, forming an angle θ. This is equivalent to a rotation around the axis of intersection of the two planes, of angle 2θ.

Further, the rotation angle is twice the angle between the two planes. This is also easy to see: let the transformation be the sequence $\mathsf{X}_2\mathsf{X}_1$ of reflections in two planes. Let us apply this transformation to the point at infinity $(\mathbf{n}_1^\mathsf{T}, 0)^\mathsf{T}$, *i.e.* the normal direction of the first plane. This is a fixed point of X_1, hence the transformation gives the point's reflection in X_2. The angle between the original point at infinity, and the transformed one, *i.e.* the fixed-axis rotation angle, is thus twice the angle between the original point at infinity and the second reflection plane. Hence, as said above, the sequence of reflections in two planes is a fixed-axis rotation, whose angle is twice the angle between the planes.

4.3 Direct Methods

The goal at hand is the computation of the coordinates of points we see reflected in an unknown specular surface. Of course, the ideal and easiest solution is to assure that the points in question are in direct view of the concerned camera. In this case, any pose estimation technique can be utilised. A potential setup is shown on figure 4.2.

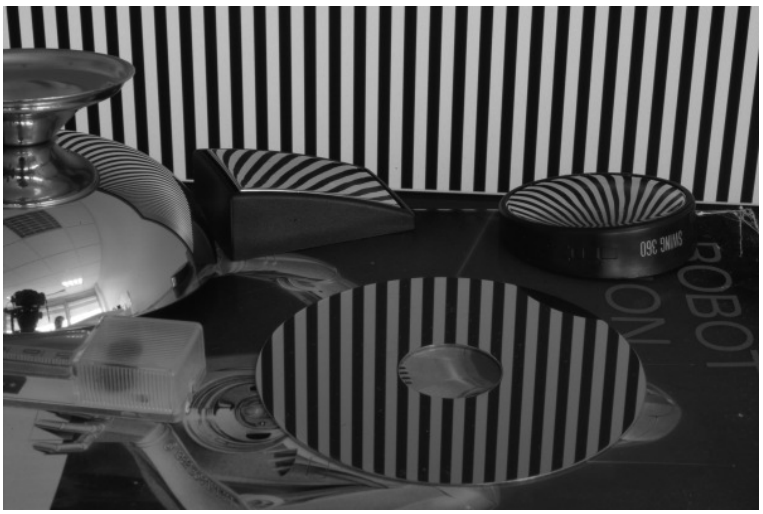


Figure 4.2: **Direct view of a calibration pattern** In this setup, the camera has a direct view of the calibration target (the LCD monitor) *and* of its reflection in the objects we wish to reconstruct.

Of course this is an ideal setup, and in most cases the position and orientation of the camera, which is fixed with respect to the specular object, produces a field of view that is insufficient to incorporate a calibration object. In fact, even in the setup shown on figure 4.2, the occupancy in the image of the target plane is close to being too small to produce stable results. Additionally, without the constraint of having a direct view of the target, we could have placed the camera in a way allowing better resolution at the specular objects, which corresponds to the goal at hand.

Another solution is to use a planar mirror that reflects the calibration object onto the camera, as illustrated in figure 4.3. Once again, using any calibration method, we obtain the pose of the reflected calibration object. What is left to determine is the pose of the planar mirror itself, which can be obtained in different ways:

- If there are known markers on the surface of the mirror, the pose of the mirror itself can be obtained from a plane-based pose estimation technique.

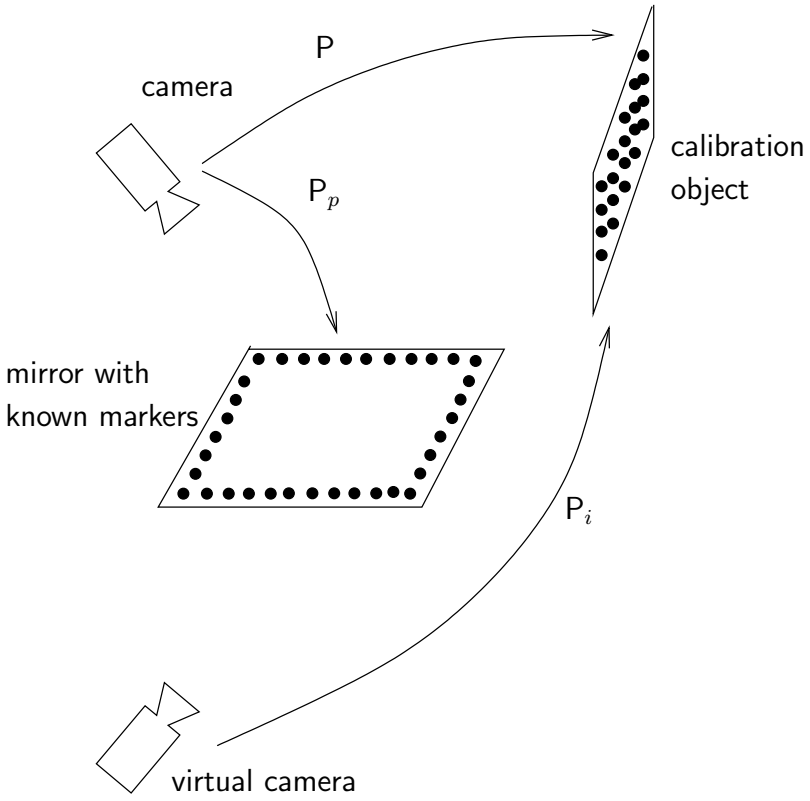


Figure 4.3: **Pose obtained by reflection by a known plane.** From the view of the known markers on the mirror plane, the pose P_p is computed. From the view of the reflection of the calibration object through the mirror plane, the virtual pose P_i is computed. The pose P of the camera is then obtained by reflecting P_i through the now known planar mirror.

- If the mirror is a disc, such as a hard drive platter, one can extract and parametrize the two ellipses in the image plane corresponding to the outer and inner rim, and use the method of [FC02] to compute the pose of the plane containing the platter.

Once the pose of the planar mirror is determined, the pose of the calibration object is trivially obtained by the method of section 4.2.1.

4.3.1 Pose alignment

In this section, we suppose that a planar target containing known reference points is rigidly attached to a pinhole camera, and that we wish to obtain the 3D coordinates of these reference points in the coordinate system of the pinhole camera. This is the setup that we used in chapter 2, a picture of which can be found on page 81. The solution we propose is an alternate application of the classical hand-eye problem [HD95], or more recently [CI01] in the case of cameras with identical projection centers.

Setup

The main device used is the combination of the digital camera and the target plane attached to it, whose reflection in the specular object is what the camera sees. For each image acquired using this device, we need to know the (relative) pose of both, camera and target plane. Since they are rigidly linked, the problem reduces to determining the pose of either one of them for the current image, and the relative pose between them, that is fixed.

We thus chose to use a fixed stereo system, placed behind the specular object and that observes the camera + target device (see figures 2.15 and 4.4). The stereo system gives us the pose of the target plane for each acquisition position (the method of [Stu00] was used), and was calibrated (intrinsics and relative pose) using directly the images of the target plane acquired during the experiments (using the methods described in [Zha00, SM99, Stu00]).

Extraction of the relative pose

The heart of the problem is the estimation of the relative position of the camera and the target plane. We solve this problem by letting the camera view a calibration object from a dozen different positions, while ensuring that the fixed target plane is in the field of view of the stereo rig. For each of the acquired images, we get:

- the pose T_i of the stereo rig with respect to the target plane fixed to the camera.

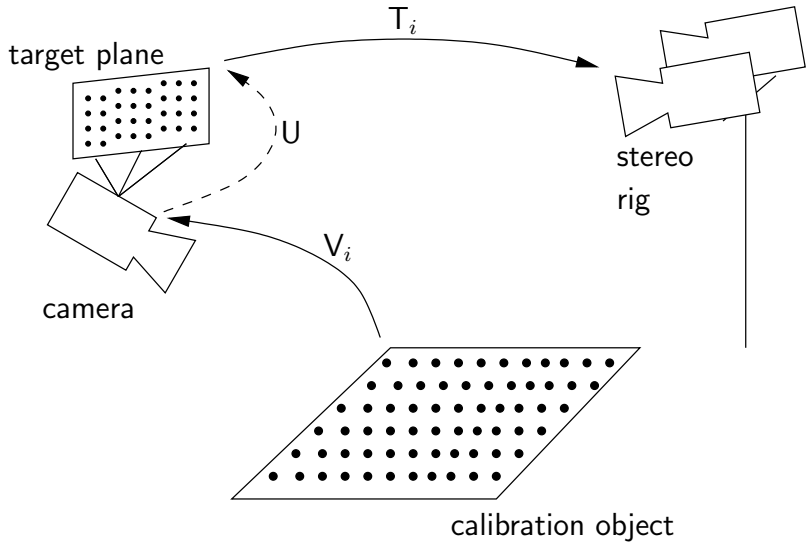


Figure 4.4: **The setup of the proposed approach.** A planar target is rigidly fixed to a camera. By displacing the camera (along with the target), a series of images are taken, in each one of which the relative pose T_i between the stereo rig and the planar target, and the relative pose V_i between the camera and the calibration object, are computed by classical pose estimation techniques. The object of this section is to compute the relative pose U between the camera and the planar target.

- the pose V_i of the camera with respect to the calibration object (the position and orientation of the calibration object is of course unknown in the reference frame of the stereo rig)

We use a modified hand-eye formulation, that states that the pose of the calibration object in the reference frame of the stereo rig is fixed and independent of the camera location:

$$\forall(i,j)\ ,\ \mathsf{V}_i\mathsf{U}\mathsf{T}_i = \mathsf{V}_j\mathsf{U}\mathsf{T}_j \tag{4.1}$$

We start by estimating the rotational part R_U of U from the rotational parts R_T and R_V of T and V:

$$\forall(i,j)\ ,\ \mathsf{R}_{\mathsf{V}i}\mathsf{R}_\mathsf{U}\mathsf{R}_{\mathsf{T}i} = \mathsf{R}_{\mathsf{V}j}\mathsf{R}_\mathsf{U}\mathsf{R}_{\mathsf{T}j}$$

which we rewrite as

$$\forall(i,j)\ ,\ \mathsf{R}_{\mathsf{V}j}^\mathsf{T}\mathsf{R}_{\mathsf{V}i}\mathsf{R}_\mathsf{U} - \mathsf{R}_\mathsf{U}\mathsf{R}_{\mathsf{T}j}\mathsf{R}_{\mathsf{T}i}^\mathsf{T} = 0_{3\times3}$$

which we solve linearly for the coefficients of R_U. Due to noise, the estimated R_U is of course not an exact rotation matrix, so we adjust it using the method of appendix B. The translational part of U is then also estimated linearly, after reinjecting the estimated R_U in equation 4.1.

4.3.2 Discussion

The approaches proposed so far all allow poses to be estimated. Nevertheless, we would prefer to obtain such results without the inherent inconveniences:

- need of planar mirror with known (or at least directly computable) pose, such as the marked mirror or hard drive platter.
- need of an additional acquisition device, namely the stereo rig

The following sections propose two methods for relaxing these constraints.

4.4 Pose Through Reflection by 3 (or more) Unknown Planes

4.4.1 Setup

We suppose a pinhole camera observing a calibration object through the reflection of a mirror we know is planar, as shown for example in figure 4.6. As the calibration object is by nature of known structure, we also suppose having a mean of computing the pose of the camera with respect to this object, when it is in direct view. For the rest of this section, we suppose without lack of generality that the calibration object is a planar target, as this is the setup with which we experimented.

We acquire images by holding a planar mirror in front of the camera in different unknown positions, such that the target plane's[1] reflection is seen by the camera. The matching approach of appendix A is used to get dense correspondances between the target plane and each image. The course of the proposed approach is to estimate the poses of the three mirror planes such that the reflected poses computed from the (reflected) views of the calibration object all correspond to a unique camera position.

We now describe how to solve the relative pose between camera and target plane, from three or more such images, or one image of three or more such mirrors.

4.4.2 Pose extraction

In the following, we adopt a global reference frame such that the target plane is at $Z = 0$, and first carry out a pose estimation for each image, as if the image were a direct view of the target plane. To be precise, we modify the method of [Stu00] by exchanging the pose's rotation matrix with a reflection matrix (we ensure its determinant equals -1). This is actually not really necessary, but correctly models that the imaging process here includes a reflection. Like in the standard case, there are two solutions for the pose; we choose the one for which the optical center has a positive Z coordinate (which corresponds to being in front of the target plane).

This procedure gives us the pose of the virtual camera that would be produced by reflecting the real camera in the planar mirror, cf. figure 4.5. If we knew the pose of the planar mirror, we could of course immediately recover the camera's true pose, as follows. Let the recovered pose of the virtual camera for image i be given via the projection matrix:

$$\mathsf{P}_i^v \sim \mathsf{S}_i \left(\mathtt{I} | -\mathbf{t}_i \right)$$

where S_i is a reflection matrix (a rotation matrix multiplied by -1), and let the associated

[1] actually an LCD monitor

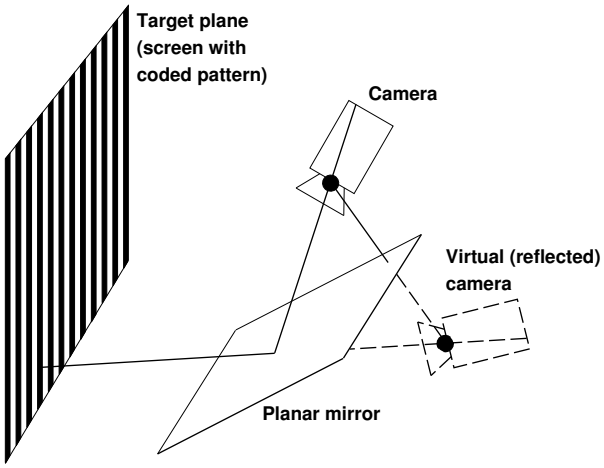


Figure 4.5: **Reflected pose.** The estimated pose of a reflected plane is equivalent to its pose viewed from a virtual reflected camera.

pose of the planar mirror be represented by homogeneous coordinates

$$\Pi_i \sim \begin{pmatrix} \mathbf{n}_i \\ d_i \end{pmatrix}$$

where we distinguish the plane's normal vector $\mathbf{n}_i$ (of unit norm), and its distance d_i from the origin.

The true camera's pose can be recovered by multiplying P_i^v with the transformation modeling the reflection in the plane Π_i:

$$\begin{aligned} \mathsf{P}_i &\sim \mathsf{P}_i^v \begin{pmatrix} \mathtt{I} - 2\mathbf{n}_i\mathbf{n}_i^\mathsf{T} & -2d_i\mathbf{n}_i \\ \mathbf{0}^\mathsf{T} & 1 \end{pmatrix} \\ &\sim \mathsf{S}_i \left(\mathtt{I} - 2\mathbf{n}_i\mathbf{n}_i^\mathsf{T} | -\mathbf{t}_i - 2d_i\mathbf{n}_i \right) \end{aligned} \tag{4.2}$$

We now have to address the question how to recover the true camera's pose without knowing the mirror planes Π_i, knowing that *with* the correct mirror positions Π_i, the camera poses P_i computed according to equation 4.2, have to be equal to one another:

$$\forall (i,j) \mathsf{P}_i \sim \mathsf{P}_j$$

To this avail, we will estimate the mirror positions Π_i that satisfy this constraint in two steps: first by estimating their orientations $\mathbf{n}_i$, and then their distances d_i. From the estimated Π_i, we then recover the pose P of the target plane as the best fit of the P_i obtained from equation 4.2.

Due to $\det\left(\mathtt{I} - 2\mathbf{n}_i\mathbf{n}_i^\mathsf{T}\right) = \det \mathsf{S}_i = -1$, we can safely eliminate the scale ambiguity in the equation $\mathsf{P}_i \sim \mathsf{P}_j$, and obtain element-wise equalities for the rotational and translational parts of the poses respectively:

$$\forall i,j : \mathsf{S}_i\left(\mathtt{I} - 2\mathbf{n}_i\mathbf{n}_i^\mathsf{T}\right) = \mathsf{S}_j\left(\mathtt{I} - 2\mathbf{n}_j\mathbf{n}_j^\mathsf{T}\right) \tag{4.3}$$

$$\forall i,j : \mathsf{S}_i\left(\mathbf{t}_i + 2d_i\mathbf{n}_i\right) = \mathsf{S}_j\left(\mathbf{t}_j + 2d_j\mathbf{n}_j\right) \tag{4.4}$$

Based on these equations, we first show how to compute the mirror plane normals $\mathbf{n}_i$, then their positions d_i.

Computing mirror plane normals $\mathbf{n}_i$.

Let $\mathsf{X}_i = \mathtt{I} - 2\mathbf{n}_i\mathbf{n}_i^\mathsf{T}$, which is of course a symmetric matrix. From (4.3), we get:

$$\mathsf{X}_i = \underbrace{\mathsf{S}_i^\mathsf{T}\mathsf{S}_j}_{\mathsf{R}_{ij}}\mathsf{X}_j \tag{4.5}$$

Furthermore, X_j is a reflection, *i.e.* $\mathsf{X}_j\mathsf{X}_j = \mathtt{I}$, therefore:

$$\mathsf{R}_{ij} = \mathsf{X}_i\mathsf{X}_j \tag{4.6}$$

Let $\mathbf{a}_{ij}$ be a vector orthogonal to $\mathbf{n}_i$ and $\mathbf{n}_j$. We therefore have:

$$\begin{aligned}
\mathsf{R}_{ij}\mathbf{a}_{ij} &= \mathsf{X}_i\mathsf{X}_j\mathbf{a}_{ij} \\
&= \left(\mathtt{I} - 2\mathbf{n}_i\mathbf{n}_i^\mathsf{T}\right)\left(\mathtt{I} - 2\mathbf{n}_j\mathbf{n}_j^\mathsf{T}\right)\mathbf{a}_{ij} \\
&= \left(\mathtt{I} - 2\mathbf{n}_i\mathbf{n}_i^\mathsf{T}\right)\mathbf{a}_{ij} \\
&= \mathbf{a}_{ij}
\end{aligned}$$

which implies that $\mathbf{a}_{ij}$ is an eigenvector to the eigenvalue 1 of R_{ij}, *i.e.* that $\mathbf{a}_{ij}$ corresponds to the rotation axis of R_{ij}.

We now have the means to compute all mirror normals $\mathbf{n}_i$, provided at least 3 mirrors are used.

1. Compute the pose equation (4.2) of all virtual cameras, as described above.
2. For all pairs of mirrors (i,j), compute R_{ij}, as per equation (4.5). Compute their eigenvectors to the eigenvalue $+1$, *i.e.* vectors $\mathbf{a}_{ij}$.
3. For every mirror i, stack all $\mathbf{a}_{ij}^\mathsf{T}$ (respectively $\mathbf{a}_{ki}^\mathsf{T}$) in a matrix A_i of size $(n-1)\times 3$ (where n is the number of mirrors). Without noise, $\mathbf{n}_i$ is orthogonal to all $\mathbf{a}_{ij}$. In the presence of noise, we perform a least squares estimation and compute $\mathbf{n}_i$ such that it minimizes $\|\mathsf{A}_i\mathbf{n}_i\|^2$ by computing $\mathbf{n}_i$ as the unit eigenvector to the smallest eigenvalue of $\mathsf{A}_i{}^\mathsf{T}\mathsf{A}_i$.

Computing mirror positions d_i.

Whereas the $\mathbf{n}_i$ are estimated one by one, we can compute the least squares solution for all the d_i at once with the linear equation system composed of one equation (4.4) per pair of mirrors. The system's design matrix is of size $3n(n-1) \times n$ and very sparse (two non-zero coefficients per row).

Computing the true camera's pose.

We now know all mirror planes Π_i, and can compute the camera pose from any one of them, according to equation (4.2). In practice, we do this computation for every mirror, and then "average" the resulting rotation matrices and position vectors that represent camera pose. We then apply a bundle adjustment style procedure for simultaneously optimizing the pose of the camera and the planar mirrors. The cost function minimized here is the reprojection error of target points, projected in the camera after reflection in the mirrors. As for the rotation component, we initialize it by computing

$$\mathsf{A} = \sum_i \mathsf{S}_i \left(\mathtt{I} - 2\mathbf{n}_i \mathbf{n}_i^\mathsf{T} \right)$$

then perform an SVD of A: $\mathsf{A} = \mathsf{U}\Sigma\mathsf{V}^\mathsf{T}$, and finally compute $\mathsf{R} = \mathsf{U}\mathsf{V}^\mathsf{T}$ (possibly followed by multiplying R with -1, to ensure it has a determinant equal to $+1$).

4.4.3 Experimental results

We used the following setup for the validation. A pinhole camera was fixed on a tripod in such a manner that its field of view contained a planar target (actually an LCD monitor) and 4 planar mirrors each reflecting an arbitrary part of the planar mirror. An image acquired with this setup is shown on figure 4.6.

The experimental validation was conducted by three different means. The first method was to pick 3 out of the 4 planar mirrors and compare the poses of the planar target when obtained with different combinations of the 3 mirrors picked. The results of this method are detailed in table 4.1. The second method was to estimate the pose of the planar target using all 4 planar mirrors (in a least squares sense), and compare the obtained pose with the pose obtained using the direct view of the planar target. These results are shown on table 4.2. Finally, the third method was to reconstruct the mirror planes with the method of chapter 3 using the indirect poses (in this case, the planar target was moved to two locations, and its pose was estimated for each case). Figure 4.7 shows the histogram of the distances of the reconstructed points with respect to the linear least squares fitted planes. Before performing the global optimization of chapter 3, all of the reconstructed points lay well under a millimeter of the fitted planes.

combination of planes	computed pose (euclidean transformation matrix)
1,2,3	$\begin{pmatrix} -1.6782\text{E}^{-02} & 9.8792\text{E}^{-01} & 1.5408\text{E}^{-01} & -1.5507\text{E}^{+02} \\ 8.8204\text{E}^{-01} & -5.7945\text{E}^{-02} & 4.6759\text{E}^{-01} & -2.8415\text{E}^{+02} \\ 4.7087\text{E}^{-01} & 1.4375\text{E}^{-01} & -8.7041\text{E}^{-01} & 6.5173\text{E}^{+02} \\ 0 & 0 & 0 & 1 \end{pmatrix}$
2,3,4	$\begin{pmatrix} -3.2672\text{E}^{-02} & 9.8890\text{E}^{-01} & 1.4492\text{E}^{-01} & -1.5316\text{E}^{+02} \\ 8.9348\text{E}^{-01} & -3.6084\text{E}^{-02} & 4.4765\text{E}^{-01} & -2.8253\text{E}^{+02} \\ 4.4791\text{E}^{-01} & 1.4411\text{E}^{-01} & -8.8239\text{E}^{-01} & 6.3801\text{E}^{+02} \\ 0 & 0 & 0 & 1 \end{pmatrix}$
1,2,4	$\begin{pmatrix} -1.4064\text{E}^{-01} & 9.7812\text{E}^{-01} & 1.5328\text{E}^{-01} & -1.1583\text{E}^{+02} \\ 8.8450\text{E}^{-01} & 5.4566\text{E}^{-02} & 4.6333\text{E}^{-01} & -2.9611\text{E}^{+02} \\ 4.4483\text{E}^{-01} & 2.0074\text{E}^{-01} & -8.7283\text{E}^{-01} & 6.4607\text{E}^{+02} \\ 0 & 0 & 0 & 1 \end{pmatrix}$
1,3,4	$\begin{pmatrix} -5.4342\text{E}^{-02} & 9.8685\text{E}^{-01} & 1.5221\text{E}^{-01} & -1.4389\text{E}^{+02} \\ 8.7280\text{E}^{-01} & -2.7101\text{E}^{-02} & 4.8732\text{E}^{-01} & -2.8732\text{E}^{+02} \\ 4.8504\text{E}^{-01} & 1.5933\text{E}^{-01} & -8.5986\text{E}^{-01} & 6.6749\text{E}^{+02} \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Table 4.1: **Pose comparisons** Matrices of the poses estimated using the proposed technique, for the same target plane, but obtained using different combinations of 3 out of the 4 planar mirrors. The translational component is expressed in millimeters. The maximal deviation from the mean translation is 2.26 centimeters. From the orientation of the target planes we can extract from the rotational component, we observe that the maximal deviation from the mean orientation is 1.65 degrees, and the variance of the orientations is 1.8.

	computed pose (euclidean transformation matrix)
direct view	$\begin{pmatrix} -6.6965\text{E}^{-02} & 9.8568\text{E}^{-01} & 1.5479\text{E}^{-01} & -1.3865\text{E}^{+02} \\ 8.8707\text{E}^{-01} & -1.8041\text{E}^{-02} & 4.9155\text{E}^{-01} & -2.881\text{E}^{+02} \\ 4.8723\text{E}^{-01} & 1.6769\text{E}^{-01} & -8.5698\text{E}^{-01} & 6.5771\text{E}^{+02} \\ 0 & 0 & 0 & 1 \end{pmatrix}$
indirect estimation	$\begin{pmatrix} -3.6843\text{E}^{-02} & 9.8691\text{E}^{-01} & 1.5701\text{E}^{-01} & -1.4854\text{E}^{+02} \\ 8.8623\text{E}^{-01} & -4.0335\text{E}^{-02} & 4.6149\text{E}^{-01} & -2.8986\text{E}^{+02} \\ 4.6179\text{E}^{-01} & 1.5615\text{E}^{-01} & -8.7314\text{E}^{-01} & 6.4973\text{E}^{+02} \\ 0 & 0 & 0 & 1 \end{pmatrix}$
indirect + direct estimation	$\begin{pmatrix} -4.4175\text{E}^{-02} & 9.9141\text{E}^{-01} & 1.2313\text{E}^{-01} & -1.4506\text{E}^{+02} \\ 8.8521\text{E}^{-01} & -1.8285\text{E}^{-02} & 4.6482\text{E}^{-01} & -2.9145\text{E}^{+02} \\ 4.6308\text{E}^{-01} & 1.2953\text{E}^{-01} & -8.7680\text{E}^{-01} & 6.6324\text{E}^{+02} \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Table 4.2: **Pose comparisons** Matrices of the poses estimated. The first row shows the result using the proposed technique with 4 planar mirrors, the second row shows the pose obtained with the direct view of the planar target, and the third row using both the planar mirrors *and* the direct view. The translational component is expressed in millimeters, therefore representing a relative error of 1.00 centimeters with respect to the direct pose. From the orientation of the target plane we can extract from the rotational component, the relative error between both estimation corresponds to 2.07 degrees.

Figure 4.6: **Experimental setup** This picture shows a view from our experimental setup. In the field of view of the camera, we see 4 planar mirrors, actually hard drive platters. At the top of the image, we have a direct view of the actual planar target. The curved mirror in the center of the image isn't used in this setup.

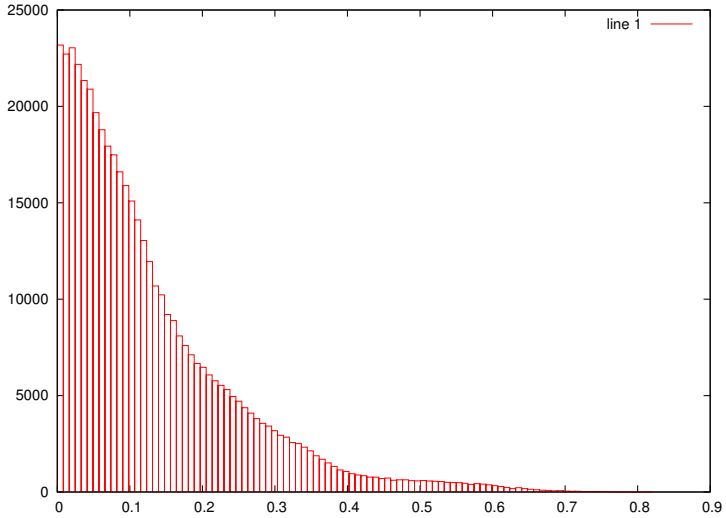


Figure 4.7: **Histogram of distances to fitted planes** Distance of the points reconstructed with the method of chapter 3 with respect to their linear least squared fitted plane (4 planes in total). Note that in this case we do not perform the global optimization of the proposed method, in order to show the quality of the estimated indirect poses. The abscissa unit represents millimeters, the ordinate represents the number of reconstructed points.

4.5 Pose estimation through the reflection of an unknown specular surface

We present the geometrical constraints that arise when an internally calibrated camera observes a target plane through the reflection by an unknown general specular surface. We show that the pose of such a target plane can be obtained from two views, when the camera is static and the target planes are placed at two different positions. These constraints can be expressed in a structure analogous to the trifocal tensor for perspective cameras, from which we present a method to linearly extract the poses of the target planes. We then discuss the degenerate configurations and the stability of the linear extraction with respect to noise.

4.5.1 Setup

Our setup is identical to that of chapter 3. Additionally, we suppose an internally calibrated *central* camera, *i.e.* a camera whose projection rays all pass through a unique point in space, the optical center. The foundation of the tensor is very simple: it relies on the fact that the light path from a feature point to the camera sensor, seen through the reflection of a mirror, is the union of two 3D lines, that intersect on the surface of the mirror, as seen on figure 4.8. The segment passing through the optical sensor and the pixel is defined by the internal camera calibration. The segment incident to the mirror point is defined by the the intersection points on the planar targets. In chapter 3, we saw that given the pose of these two target planes, we are able to reconstruct points of the specular surface. In this section, by using the known local coordinates of corresponding points in the target planes, we study the recovery of the two poses of the target planes with respect to the camera's optical center. We show that by constraining the two 3D lines to intersect, there exists a trilinear tensor in terms of the parameters of the plane poses, and that these terms can be uniquely extracted given 26 matches between camera pixels and their corresponding feature points in the target planes.

The summary of the requirements follows:

- An internally calibrated central camera.
- A planar target placed at two different unknown locations such that its reflection in the specular surface is seen by the camera.
- Matching between camera pixels and the coordinates of their origin in the target plane frames (see appendix A).

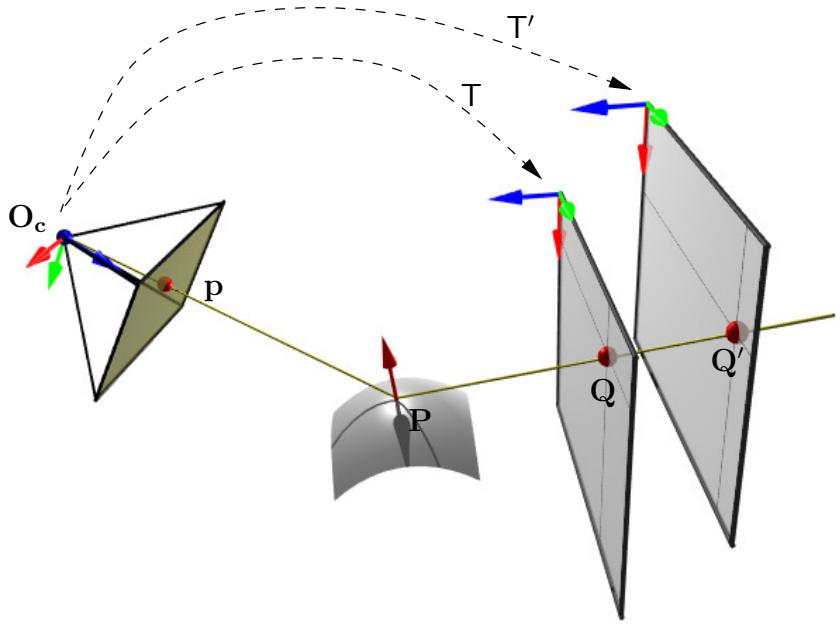


Figure 4.8: **Image Formation.** A pixel in the camera image corresponds to a known 3D line. Image based techniques allow us to determine the coordinates of its corresponding features $\mathbf{Q}$ and $\mathbf{Q}'$ for both planar targets. The goal at hand is to determine the two transformations T and T′.

4.5.2 Tensor Formation

We consider a simple central camera with a known projection function, and whose projection center $\mathbf{O}_c = (0, 0, 0, 1)^\mathsf{T}$ is without loss of generality the origin of our reference coordinate system. The type of projection of the camera is not relevant, apart from the fact that it must be central and that we must have a means to compute projection rays given pixel coordinates.

Let us consider a pixel $\mathbf{p} = (p_1, p_2, p_3)^\mathsf{T}$, which can be back projected to a point $\mathbf{d} = (d_1, d_2, d_3, d_4)^\mathsf{T}$ on the corresponding projection ray by using the camera's inverse projection function (any point on the ray can be taken here). As we have supposed a known matching between pixels and points on the target planes, we know that $\mathbf{p}$ is the image of the reflection of the point $\mathbf{Q} = (Q_1, Q_2, 0, Q_4)^\mathsf{T}$ in the first target plane, and of the point $\mathbf{Q}' = (Q'_1, Q'_2, 0, Q'_4)^\mathsf{T}$ in the second one (expressed in coordinates local to the planes). Due to the way the image is formed, we know that the specular surface lies at the intersection of the line spanned by $\mathbf{Q}$ on the first target plane and $\mathbf{Q}'$ on the

second, and the line passing by the optical center and the corresponding pixel, as shown on figure 4.8.

Let R and $\mathbf{t} = (t_1, t_2, t_3)^\mathsf{T}$ (*resp.* R′ and $\mathbf{t}'$) be the rotation matrix and translation vector of the transformation T from the first (*resp.* second, T′) target plane reference coordinate system to the camera reference coordinate system:

$$\mathsf{T} = \begin{pmatrix} \mathsf{R} & \mathbf{t} \\ \mathbf{0}^\mathsf{T} & 1 \end{pmatrix} \ , \ \mathsf{T}' = \begin{pmatrix} \mathsf{R}' & \mathbf{t}' \\ \mathbf{0}^\mathsf{T} & 1 \end{pmatrix}$$

Constraining the 3D lines $(\mathbf{O}_c, \mathbf{d})$ and $(\mathsf{T}\mathbf{Q}, \mathsf{T}'\mathbf{Q}')$ to intersect can be expressed as a coplanarity constraint between $\mathbf{O}_c$, $\mathbf{d}$, $\mathsf{T}\mathbf{Q}$, and $\mathsf{T}'\mathbf{Q}'$, which algebraically corresponds to:

$$\begin{vmatrix} 0 & d_1 & & \\ 0 & d_2 & \mathsf{T}\mathbf{Q} & \mathsf{T}'\mathbf{Q}' \\ 0 & d_3 & & \\ 1 & d_4 & & \end{vmatrix} = 0 \tag{4.7}$$

after noting

$$\mathbf{q} = \begin{pmatrix} Q_1 \\ Q_2 \\ Q_4 \end{pmatrix}, \mathbf{q}' = \begin{pmatrix} Q'_1 \\ Q'_2 \\ Q'_4 \end{pmatrix}$$

this simplifies to ($\overline{\mathsf{R}}$ denotes the first two columns of R):

$$\begin{vmatrix} d_1 & & & \\ d_2 & \left(\overline{\mathsf{R}}_{3\times 2} \quad \mathbf{t}\right)\mathbf{q} & \left(\overline{\mathsf{R}'}_{3\times 2} \quad \mathbf{t}'\right)\mathbf{q}' \\ d_3 & & & \end{vmatrix} = 0 \tag{4.8}$$

We can factor out the terms of $\mathbf{d}$, $\mathbf{q}$ and $\mathbf{q}'$ and rewrite equation 4.8 as a trilinear constraint:

$$\sum_{i,j,k=1..3} d_i q_j q'_k \mathcal{T}_{ijk} = 0 \tag{4.9}$$

The trilinear tensor $\mathcal{T}_{ijk}$ only depends on the pose parameters R, R′, $\mathbf{t}$ and $\mathbf{t}'$ as explained in the following section. By using 26 or more point correspondences, a linear least squares solution to the estimation of the $\mathcal{T}_{ijk}$ of equation 4.9 can be found up to scale by SVD, in terms of d_i, q_j and q'_k (see table 4.3).

4.5.3 Extraction of (R, $\mathbf{t}$) and (R′, $\mathbf{t}'$)

The tensor does not know about rotations or translations, so they have to be extracted from the solution. We denote the i-th column of R by $\mathbf{r_i}$, the first two columns of R by $\overline{\mathsf{R}}$, and the matrix representing the cross product by vector $\mathbf{x}$ by $[\mathbf{x}]_\times$. The following

constraints are obtained by rearranging the equations of table 4.3, and show that the parameters of the poses are in terms of cross-products of the computed tensor (up to scale):

$$\begin{pmatrix} \mathcal{T}_{131} & \mathcal{T}_{132} \\ \mathcal{T}_{231} & \mathcal{T}_{232} \\ \mathcal{T}_{331} & \mathcal{T}_{332} \end{pmatrix} \sim [\mathbf{t}]_{\times} \overline{\mathsf{R}'}$$

$$\begin{pmatrix} \mathcal{T}_{113} & \mathcal{T}_{123} \\ \mathcal{T}_{213} & \mathcal{T}_{223} \\ \mathcal{T}_{313} & \mathcal{T}_{323} \end{pmatrix} \sim -[\mathbf{t}']_{\times} \overline{\mathsf{R}}$$

$$\begin{pmatrix} \mathcal{T}_{111} \\ \mathcal{T}_{211} \\ \mathcal{T}_{311} \end{pmatrix} \sim \mathbf{r_1} \times \mathbf{r'_1}$$

$$\begin{pmatrix} \mathcal{T}_{112} \\ \mathcal{T}_{212} \\ \mathcal{T}_{312} \end{pmatrix} \sim \mathbf{r_1} \times \mathbf{r'_2}$$

$$\begin{pmatrix} \mathcal{T}_{121} \\ \mathcal{T}_{221} \\ \mathcal{T}_{321} \end{pmatrix} \sim \mathbf{r_2} \times \mathbf{r'_1}$$

$$\begin{pmatrix} \mathcal{T}_{122} \\ \mathcal{T}_{222} \\ \mathcal{T}_{322} \end{pmatrix} \sim \mathbf{r_2} \times \mathbf{r'_2}$$

$$\begin{pmatrix} \mathcal{T}_{133} \\ \mathcal{T}_{233} \\ \mathcal{T}_{333} \end{pmatrix} \sim \mathbf{t} \times \mathbf{t}'$$

The Rotational Component From the preceding constraints, we obtain:

$$\mathbf{r_1} \sim \begin{pmatrix} \mathcal{T}_{111} \\ \mathcal{T}_{211} \\ \mathcal{T}_{311} \end{pmatrix} \times \begin{pmatrix} \mathcal{T}_{112} \\ \mathcal{T}_{212} \\ \mathcal{T}_{312} \end{pmatrix}$$

$$\mathbf{r_2} \sim \begin{pmatrix} \mathcal{T}_{121} \\ \mathcal{T}_{221} \\ \mathcal{T}_{321} \end{pmatrix} \times \begin{pmatrix} \mathcal{T}_{122} \\ \mathcal{T}_{222} \\ \mathcal{T}_{322} \end{pmatrix}$$

$$\mathbf{r'_1} \sim \begin{pmatrix} \mathcal{T}_{111} \\ \mathcal{T}_{211} \\ \mathcal{T}_{311} \end{pmatrix} \times \begin{pmatrix} \mathcal{T}_{121} \\ \mathcal{T}_{221} \\ \mathcal{T}_{321} \end{pmatrix}$$

$$\mathbf{r'_2} \sim \begin{pmatrix} \mathcal{T}_{112} \\ \mathcal{T}_{212} \\ \mathcal{T}_{312} \end{pmatrix} \times \begin{pmatrix} \mathcal{T}_{122} \\ \mathcal{T}_{222} \\ \mathcal{T}_{322} \end{pmatrix}$$

Here, the columns of R and R′ are obtained up to scale, as we only obtain their direction given by a cross product. The unknown scale can however be recovered up to sign, as the columns of R and R′ have to be of norm 1 by construction. $\mathbf{r_3}$ (*resp.* $\mathbf{r'_3}$) is then obtained by cross product of $\mathbf{r_1}$ and $\mathbf{r_2}$ (*resp.* $\mathbf{r'_1}$ and $\mathbf{r'_2}$).

There is only one sign ambiguity, that can be explained geometrically: the symmetric poses with respect to the camera's optical center are evidently a correct solution to the intersection constraint, and can be easily eliminated *a priori*, for example by supposing the mirror surface is in front of the camera.

In practice, R and R′ are then adjusted to represent exact rotation matrices (see appendix B).

The Translational Component We obtain $\mathbf{t}$ and $\mathbf{t}'$ with a similar method:

$$\mathbf{t} \sim \begin{pmatrix} \mathcal{T}_{131} \\ \mathcal{T}_{231} \\ \mathcal{T}_{331} \end{pmatrix} \times \begin{pmatrix} \mathcal{T}_{132} \\ \mathcal{T}_{232} \\ \mathcal{T}_{332} \end{pmatrix} \quad \mathbf{t}' \sim \begin{pmatrix} \mathcal{T}_{113} \\ \mathcal{T}_{213} \\ \mathcal{T}_{313} \end{pmatrix} \times \begin{pmatrix} \mathcal{T}_{123} \\ \mathcal{T}_{223} \\ \mathcal{T}_{323} \end{pmatrix}$$

The computed values are only the directions of the translation vectors, but their true scale can be obtained by re-injecting the computed directions in the constraints shown above, and scaling accordingly. Here we do not have any sign ambiguity, as it was ruled out in the previous paragraph.

4.5.4 Degenerate Configurations

We now study the configurations in which $\mathcal{T}_{ijk}$ cannot be estimated, and therefore a solution to the pose estimations can not be extracted.

Planar Mirror

When the mirror surface is a plane there exists an homography between the pixel coordinates and the target coordinates, as this configuration is equivalent to having a direct view of the target planes, but viewed from the camera reflected by the planar mirror. Here, the tensor has no geometric background, and as such cannot be estimated.

This case is easy to rule out, as it is sufficient to test whether we have an homography between camera pixels and target features.

Surfaces of Revolution

If the specular surface is a surface of revolution and if the optical center of the camera lies on the axis of revolution, then the tensor can not be estimated uniquely using the system of linear equations 4.9. To explain this, we first give a geometric interpretation to the column vectors in the determinant of equation 4.8. Let us call them $\mathbf{d}, \mathbf{s}$ and $\mathbf{s}'$. These vectors were obtained by eliminating the last homogeneous coordinate of three 3D points of equation 4.7 (one on a projection ray of the camera and two on the target planes). Since the camera's optical center is the origin in our formulation, these vectors $\mathbf{d}, \mathbf{s}$ and $\mathbf{s}'$ correspond to the points at infinity of the lines joining the optical center with these three 3D points. The vanishing of the determinant means that the three points at infinity are collinear.

Let us now consider the case of a surface of revolution and of the camera's optical center lying on the axis of revolution. Let $\mathbf{a}$ be the three coordinates of the revolution axis' point at infinity. Consider any pixel and its associated projection ray. The projection ray contains the optical center and is thus coplanar with the axis of revolution. Hence, the ray reflected in the surface of revolution, is also coplanar with the axis of revolution, and the original projection ray. This implies that the 3D points of equation 4.7 are all coplanar with the axis of revolution. Consequently, the three points at infinity explained above, are collinear with the point at infinity $\mathbf{a}$ of the axis of revolution.

Based on these observations, we now explain that the tensor does not have a unique solution, if estimated via a linear equation system. Let T and T′ be the correct solution of the target planes' poses, implying that the three points at infinity of $\mathbf{d}, \mathbf{s}$ and $\mathbf{s}'$ are collinear. Due to above observations, they are in addition collinear with $\mathbf{a}$. Let M and M′ be any two 3×3 projective transformations that preserve all lines through $\mathbf{a}$ (there are three degrees of freedom for such transformations). Obviously, $\mathbf{a}, \mathbf{d}, \mathrm{M}\mathbf{s}$ and $\mathrm{M}'\mathbf{s}'$

will be collinear. Hence, the transformations

$$\mathsf{M}\begin{pmatrix}\overline{\mathsf{R}} & \mathbf{t}\end{pmatrix}$$
$$\mathsf{M}'\begin{pmatrix}\overline{\mathsf{R}'} & \mathbf{t}'\end{pmatrix}$$

are also a valid solution for all equations 4.8. These transformations do not in general satisfy orthonormality constraints inherited by R and R′; however, with linear equations only, orthonormality constraints are not enforced. In conclusion, the linear equation system on the tensor will provide ambiguous solutions (at least 6 degrees of ambiguity, for M and M′).

The case of a surface of revolution could actually be solved, by explicitly taking into account the specific characteristics of it. This is not developed further, as this case is rather marginal.

4.5.5 (In)Stability with respect to noise

Whereas the specular tensor is an elegant theoretical approach to the pose of a reflected target plane, our experiments on simulation data show that it is far too unstable to be used in real world setups. In a simulated typical camera setup, the pose parameters can only be correctly extracted if the noise in the correspondences in the target plane are perfect, which is of course not realistic due to image based feature extraction, and internal camera calibration. Figure 4.9 shows a plot of the condition number given by the SVD of $\mathcal{T}_{ijk}$, and clearly shows that even with minimal noise, a solution to equation 4.9 cannot be estimated.

4.5.6 Discussion

We have studied the theoretical constraints that arise when a calibrated camera observes a target plane placed at two different unknown positions, through the reflection by an unknown general specular surface. We have shown that with perfect data, the poses of the planes can be recovered linearly, and as such that the specular surface can be reconstructed. We were very excited when we first discovered the possible constraints, as a robust pose recovery would have lead to an automatic and practical specular surface inspection method, but the extreme instability that arose from our simulations was disappointing.

The tensor we describe is very similar to the dual homography tensor of [SW00], for the computation of three camera poses observing coplanar points, when the points are allowed to move on straight lines between views. We can also find many resemblances with the general camera calibration of [SR04]: in the general case where the camera rays are unconstrained, pose parameters are extracted using the collinearity of points

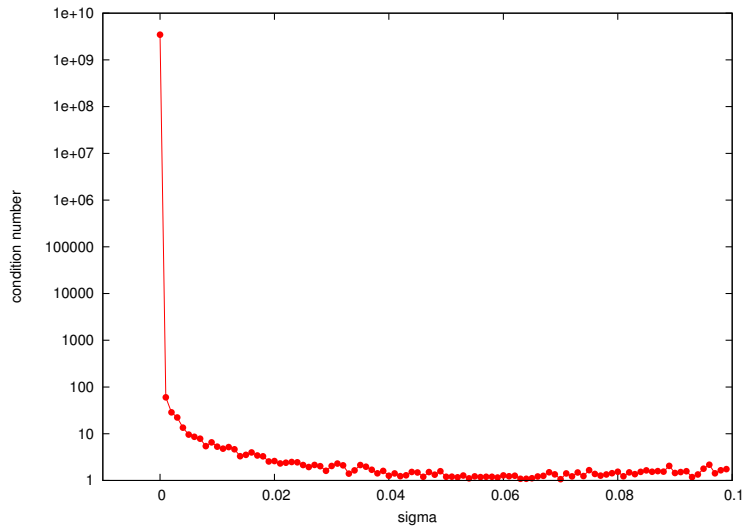


Figure 4.9: **Instability with respect to noise** Plot of the condition number given by the SVD of $\mathcal{T}_{ijk}$ of equation 4.9 (see appendix B). The setup consisted of a 1000 randomly chosen pixels reflected through an arbitrary specular surface to compute the correspondances in 2 planes. The abscissa is the variance of the gaussian noise we added to the correspondances in the planes.

from 3 planar targets, whereas we work with 2 target planes, but with the additional constraints that all camera projection rays pass through the optical center and that the internal parameters of the camera are known.

4.6 Conclusion

We have proposed several methods for obtaining the pose of a calibration object or target plane when it is not in direct view of a camera. Applications for such a scenario include most existing specular surface reconstruction methods, and could also encompass wide baseline multi-camera setups without a common field of view. The first methods require additional objects or cameras, and as such are not very flexible.

On the other hand the method requiring 3 planar mirrors is trivial to utilize, as it only requires the view of the calibration object through these mirrors, placed at any convenient locations. One could imagine a device consisting of 3 or more articulated mirrors that could be placed in front of a camera to calibrate, in order to obtain its internal and external calibration without any constraint on its orientation, allowing virtually any setup for a multi-camera aquisition system.

Finally, we presented what first seemed to be a universal reconstruction technique, or conversely pose estimation technique through the reflection of an arbitrary mirror. The extreme instability that arose is of course extremely disappointing, and may reveal what seems to be a practical limit in terms of flexible setups for specular surface reconstruction.

$\mathcal{T}_{ijk}$	tensor term	tensor factor
$\mathcal{T}_{111}$	$d_1 q_1 q'_1$	$R_{21}R'_{31} - R_{31}R'_{21}$
$\mathcal{T}_{112}$	$d_1 q_1 q'_2$	$R_{21}R'_{32} - R_{31}R'_{22}$
$\mathcal{T}_{113}$	$d_1 q_1$	$R_{21}t'_3 - R_{31}t'_2$
$\mathcal{T}_{121}$	$d_1 q_2 q'_1$	$R_{22}R'_{31} - R_{32}R'_{21}$
$\mathcal{T}_{122}$	$d_1 q_2 q'_2$	$R_{22}R'_{32} - R_{32}R'_{22}$
$\mathcal{T}_{123}$	$d_1 q_2$	$R_{22}t'_3 - R_{32}t'_2$
$\mathcal{T}_{131}$	$d_1 q'_1$	$t_2 R'_{31} - t_3 R'_{21}$
$\mathcal{T}_{132}$	$d_1 q'_2$	$t_2 R'_{32} - t_3 R'_{22}$
$\mathcal{T}_{133}$	d_1	$t_2 t'_3 - t_3 t'_2$
$\mathcal{T}_{211}$	$d_2 q_1 q'_1$	$R_{31}R'_{11} - R_{11}R'_{31}$
$\mathcal{T}_{212}$	$d_2 q_1 q'_2$	$R_{31}R'_{12} - R_{11}R'_{32}$
$\mathcal{T}_{213}$	$d_2 q_1$	$R_{31}t'_1 - R_{11}t'_3$
$\mathcal{T}_{221}$	$d_2 q_2 q'_1$	$R_{32}R'_{11} - R_{12}R'_{31}$
$\mathcal{T}_{222}$	$d_2 q_2 q'_2$	$R_{32}R'_{12} - R_{12}R'_{32}$
$\mathcal{T}_{223}$	$d_2 q_2$	$R_{32}t'_1 - R_{12}t'_3$
$\mathcal{T}_{231}$	$d_2 q'_1$	$t_3 R'_{11} - t_1 R'_{31}$
$\mathcal{T}_{232}$	$d_2 q'_2$	$t_3 R'_{12} - t_1 R'_{32}$
$\mathcal{T}_{233}$	d_2	$t_3 t' x - t_1 t' z$
$\mathcal{T}_{311}$	$d_3 q_1 q'_1$	$R_{11}R'_{21} - R_{21}R'_{11}$
$\mathcal{T}_{312}$	$d_3 q_1 q'_2$	$R_{11}R'_{22} - R_{21}R'_{12}$
$\mathcal{T}_{313}$	$d_3 q_1$	$R_{11}t'_2 - R_{21}t'_1$
$\mathcal{T}_{321}$	$d_3 q_2 q'_1$	$R_{12}R'_{21} - R_{22}R'_{11}$
$\mathcal{T}_{322}$	$d_3 q_2 q'_2$	$R_{12}R'_{22} - R_{22}R'_{12}$
$\mathcal{T}_{323}$	$d_3 q_2$	$R_{12}t'_2 - R_{22}t'_1$
$\mathcal{T}_{331}$	$d_3 q'_1$	$t_1 R'_{21} - t_2 R'_{11}$
$\mathcal{T}_{332}$	$d_3 q'_2$	$t_1 R'_{22} - t_2 R'_{12}$
$\mathcal{T}_{333}$	d_3	$t_1 t'_2 - t_2 t'_1$

Table 4.3: **The Specular Tensor.** Each element $\mathcal{T}_{ijk}$ of the specular tensor is in terms of the rotations and translations of the target planes with respect to the camera projection center. For clarity, we have scaled $\mathbf{q}$ and $\mathbf{q}'$ such that $q_3 = q'_3 = 1$.

5.

Conclusion and Perspectives

In this thesis we have studied the reconstruction of specular surfaces from images. This subject has received relatively little attention from the computer vision community, which in turn implies that these types of surfaces are usually discarded as noise. We have presented two different methods for recovering the shape of an arbitrary specular surface, from the reflection of known scene features. A ready-to-use implementation of the second method was tested in a real-world setup. With respect to the state of the art, these methods do not make the usual assumption of surface smoothness or even continuity.

The method presented first extends the well known framework of space carving [KS00] to incorporate surfaces with a sharp specular component in their reflectance. Instead of using photometric invariance with respect to viewpoint, we proposed to use geometric invariance, namely by observing surface orientation rather than surface color. However the required setup is quite cumbersome in the sense that the camera must be moved to several locations, of which the poses must be determined. Furthermore, the requirement of having known correspondences between pixels and scene points practically requires either having a large calibrated environment, for example a hemispheric dome over the object, or moving a target to different locations, thus adding an additional source of noise.

Concerning the accuracy of the method, a rather deceiving observation should be made. The field of applications of specular surface reconstruction techniques is very narrow, objectively only limited to metrology and industrial inspection. In effect, in domains where accuracy is not primordial, such as virtual reality or computer graphics, chances are these types of surfaces will rarely occur, and if ever they do, traditional methods such as contact probing or laser scanning coupled with a means to give a matte finish to the surface will prevail. As such, the lack of precision of space carving approaches coupled with the rather complex and constrained setup that is required very probably imply that this method will have very few practical applications.

The triangulation approach we presented in chapter 3 relaxes a large number of constraints of the previous method, namely by only requiring two views from a fixed camera viewpoint. The results we have obtained seem very promising with respect to the accuracy that was obtained, and the relatively unconstrained setup that is needed. In a very constrained scenario such as industrial inspection, with precisely known camera calibration and target planes accurately placed by robotic manipulators, 3D models of reflecting surfaces could easily be recovered. Work has still to be done concerning the matching process between camera pixels and points from the target planes, as in these types of applications the accuracy and speed of such a task are primordial. The approach we propose in appendix A concerning this problem is satisfactory accuracy-wise, but maybe too time consuming for real-time inspections on a production chain.

Chapter 4 proposed several methods for the computation of the pose of a plane (or more generally a calibration object) when this plane is not in the field of view of the camera. A method close to the classical hand-eye problem was proposed, with the obvious inconvenience of requiring an additional imaging device.

A method using the reflection of the plane in mirrors we know are planar may allow more flexible setups, not only in the field of specular surface reconstruction, but also whenever multiple cameras without a common field of view are required. Additionally, the fact that the target plane be reflected by 3 planar mirrors actually is equivalent to having 3 different views of it, which means that the internal calibration of the camera can be computed without additional intervention.

Finally, geometrical constraints that could have lead to uncalibrated specular surface reconstruction, or conversely indirect pose estimation through the reflection of an arbitrary mirror, were investigated. The extreme instability that occurred was in this sense very disappointing, and may reveal what seems to be a practical limit in terms of flexible setups for specular surface reconstruction.

Additional work would probably be welcome in the field of precise matching using structured light approaches, with a strong constraint on acquisition times. Additionally, due to diverse compressions and magnifications appearing respectively on convex and concave parts of specular objects, an automatic approach with varying spatial resolution would seem interesting.

Concerning specular surface reconstruction, methods that do not require calibrated scene points would be an obligatory starting point before obtaining flexible reconstruction algorithms able to treat surfaces with arbitrary reflectances. This can probably only be done by integrating additional cues such as silhouettes, or having strong priors on the geometry of the object and/or environment.

A.

Subpixel matching

The task of matching camera pixels with either their direct or reflected source in the surrounding scene is both a vital and arguably uninteresting task. While manually clicking and associating points is an option when there are only a few images and a few feature points, in our case we need a solution that finds a correspondence for every pixel in the region of interest of our image, *i.e.* in the order of tens or hundreds of thousands. A typical approach in this situation is to use color coded structured light, in our case using binary patterns, of which an extensive review can be found in [SPB04].

Our setup uses an LCD monitor whose display is seen reflected by the specular surfaces onto the image plane of the camera. The goal of the matching is to obtain correspondences between pixel coordinates in the camera image with their source pixel on the monitor. A straightforward method, illustrated in table A.1 is to display a series of binary patterns on the monitor which encode the binary form of the coordinate of each pixel.

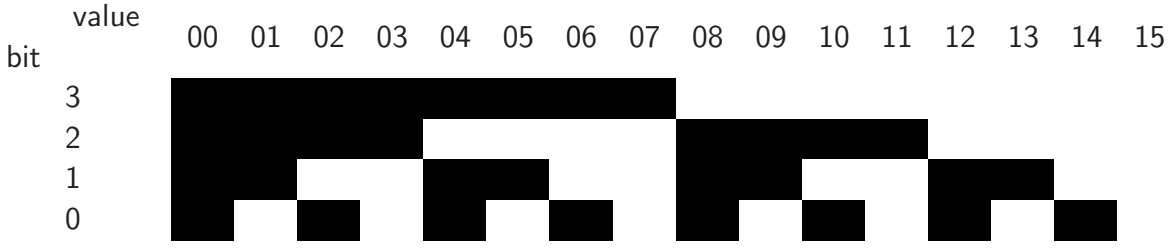


Table A.1: **Binary coding** - abscissa represents the integer value to encode. For each of these values, the column corresponds to the series of colors (in our case just black or white) that must encode it. For example, the value "7" will be encoded by the following sequence: "black - white - white - white". If there is an error in determining the color of bit 3, thus reading "white - white - white - white", the decoded value will be 15.

These binary codes of course encode a one-dimensional value, so two series of patterns need to be displayed to encode the x and y coordinate of each pixel. The number of images that need to be displayed for the matching of a 2^n by 2^n image is therefore $2n$. While these $2n$ images are the minimum required, making the decision, given an image pixel intensity, whether the screen source pixel is black or white is actually a difficult task, as shown on figure A.1:

- in areas where the black and white stripes in the image are contiguous, the imaged corresponding pixel is seen as different shades of gray
- due to LCD monitor imperfection, white stripes tend to *bleed* over the black stripes, again producing shades of gray.

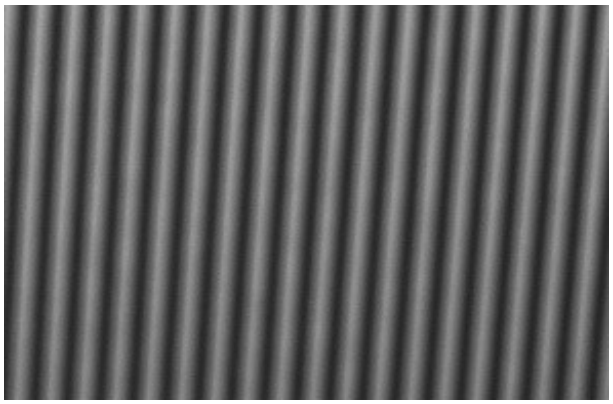


Figure A.1: **Typical binary code capture**

To overcome the preceding limitations we apply the following method:

- We use Gray coding [Gra53] instead of raw binary codes, as in the latter case the error in discriminating a black from white pixel of a high order bit in the encoded value causes a large disparity in this decoded value (see table A.2). Gray codes on the other hand are designed in such a manner that two adjacent values only differ by one bit, thus implying that the flip in a single bit will only cause a unit error. Implementations of the conversion from regular values to Gray codes are widely found, for example in [PFTV92].
- Instead of using one image for every bit, we use the method proposed by [SS03], and display two images: the first image is the original Gray code, and the second one is the inverse. In this setup, for each pixel we only have to compare intensities locally: if the pixel in the first image is brighter than in the second, then the value of the pixel is "white", and inversely.

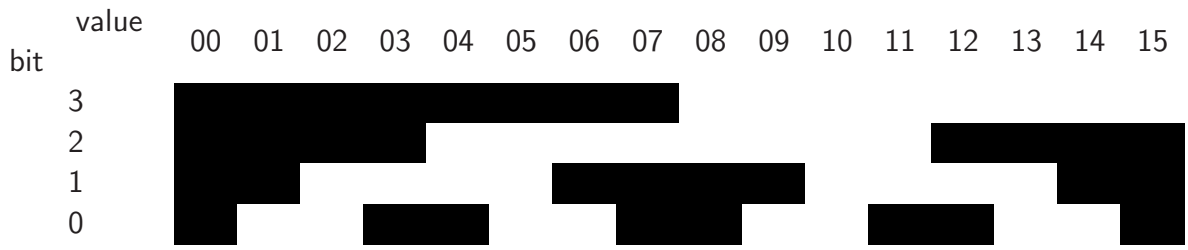
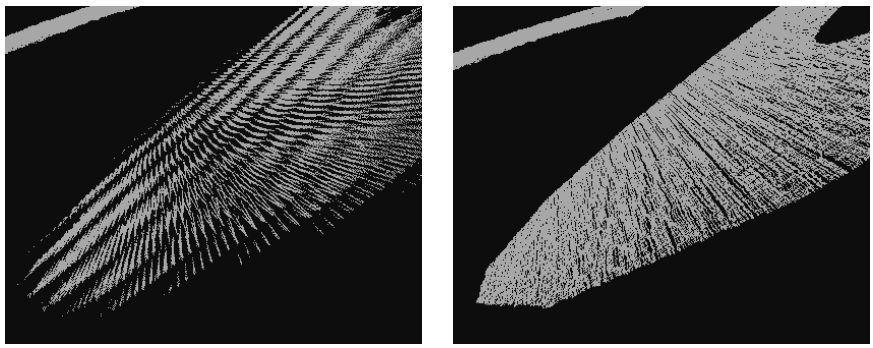


Table A.2: **Gray codes.**Error in extracting the value for one bit causes only a unit error

The raw decoding of the Gray code images assigns an integer 2D coordinate to each pixel in the region of interest of the camera. Depending on the resolution of the image sensor, and/or due to the apparent magnification that occurs at concave locations on the specular surface, this means that neighboring image pixels can be assigned with the same screen correspondence. While these raw matches already provide high quality results, the matching of several image pixels with the same source on the screen induces some geometric error that causes visible artifacts in the reconstructed surface for example when using the triangulation method of chapter 3, as can be seen on figure A.2. In order to obtain high quality matches, we would like to obtain subpixel matches, that is to say that for every pixel in the camera image, the matched point is the real (*i.e.* floating point) coordinate in the screen reference system.



without smoothing with smoothing

Figure A.2: **Reconstruction of a planar mirror** Side view of the reconstruction of a planar surface with and without smoothing the raw decoded values from the Gray codes.

We propose an energy minimization method that will allow us to smooth the decoded coordinates, but with the additional constraint that the smoothed coordinates still correspond to the input images.

Let $u(x, y)$ and $v(x, y)$ denote the coordinates of the target point corresponding to the camera pixel (x, y). Instead of directly smoothing u and v as in [SS03], we use an

energy minimization approach to ensure that the smoothed correspondences will still link camera pixels with their corresponding origin on the target planes.

We minimize the following energy functional with respect to u and v:

$$\begin{aligned} E(u,v) &= \sum_k \int_\Omega (\mathcal{G}_k(u,v) - \mathcal{I}_k(x,y))^2 \, dx \, dy \\ &+ \lambda \int_\Omega |\nabla u|^2 + |\nabla v|^2 \, dx \, dy \end{aligned}$$

where Ω is the mirror image region, $\mathcal{G}_k$ are the displayed Gray code images and $\mathcal{I}_k$ are the images captured by the camera.

The first energy term is the data term. It penalizes correspondences for which the color $\mathcal{I}_k(x,y)$ captured by the camera and its corresponding Gray code $\mathcal{G}_k(u,v)$ are not the same. We first scale the camera images intensities pixel-wise, so that 0 and 1 intensities correspond to pure black and pure white. This referential is computed by displaying entirely black and entirely white images on the planar targets. For non-integer values of u and v, $\mathcal{G}_k(u,v)$ is computed using bilinear interpolation.

The second term is a homogeneous regularizer. It penalizes large variations on the correspondence functions. The λ parameter sets the compromise between data evidence and smoothing.

The energy functional is minimized by a steepest descent. The descent direction is given by the Euler-Lagrange equations,

$$\frac{\partial u_i}{\partial t} = -\sum_k 2(\mathcal{G}_k - \mathcal{I}_k)\frac{\partial \mathcal{G}_k}{\partial u_i} + \lambda \, 2\Delta u_i$$

for $u_1 = u$ and $u_2 = v$.

Note that the Gray code images projected on the screen serve a double purpose: they are used "traditionally" to provide an initialization to the energy minimization function, and they are also used to compute the data term $\mathcal{G}_k(u,v) - \mathcal{I}_k(x,y)$. However this data term could be computed from any series of images containing sufficient texture.

A qualitative result showing the effectiveness of the method can be seen on figure A.3.

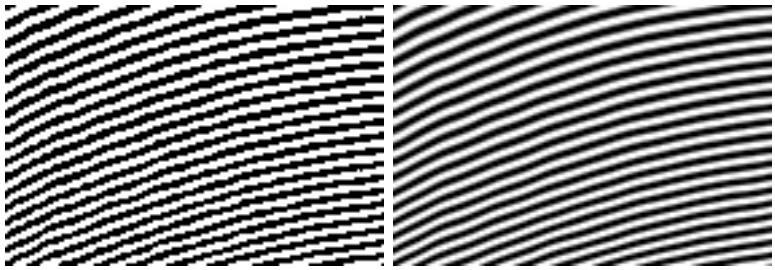


Figure A.3: **Texture mapping using the computed correspondences - before and after the refinement step** The computed correspondences are used as a lookup-table for texturing the camera image from an image defined in the screen coordinate system. We calculate brightness values for each pixel by using bilinear interpolation around the computed match.

B.

Singular Value Decomposition

A general $m \times n$ matrix A has a singular value decomposition (SVD) [PFTV92]:

$$\underset{m\times n}{\begin{pmatrix} & \mathsf{A} & \\ & & \end{pmatrix}} = \underset{m\times n}{\begin{pmatrix} & \mathsf{U} & \\ & & \end{pmatrix}} \underbrace{\underset{n\times n}{\begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{pmatrix}}}_{\Sigma} \underset{n\times n}{\begin{pmatrix} & \mathsf{V}^{\mathsf{T}} & \\ & & \end{pmatrix}}$$

where U is a $m \times n$ orthogonal matrix, Σ is a diagonal $n \times n$ matrix, and V an $n \times n$ orthogonal square matrix. The values $\sigma_i = \Sigma_{ii}$ of the diagonal matrix Σ are called *singular values*, and we suppose without lack of generality that they are sorted in non-increasing order:

$$\sigma_1 \geq \sigma_2 \geq \ldots \sigma_n \geq 0$$

This decomposition is unique up to permutations of the columns of U and V and elements of Σ, and linear combinations of the columns i, j in U and V when $\sigma_i = \sigma_j$.

Singular value decomposition has many practical applications:

- **The condition number** of matrix A can be defined by the ratio of the largest to smallest singular value $\frac{\sigma_1}{\sigma_n}$. The matrix A is singular if the condition number is infinite, and ill-conditioned if the condition number approaches machine floating-point precision.

- In the case of a linear mapping $\mathbb{R}^m \rightarrow \mathbb{R}^n$ defined by matrix A:

  $$\mathsf{A}\mathbf{x} = \mathsf{B}$$

  with A which is singular, then the subspace of $\mathbb{R}^m$ such that $\mathsf{A}\mathbf{x} = \mathbf{0}_n$ is determined by the columns of V corresponding to the singular values that are zero. This property is used extensively in most linear least-squares fitting tasks.

Applications

Closest rotation matrix

Consider a matrix $\bar{\mathsf{R}}$ linearly estimated as a rotation from a set of equations. In general, this estimation does not enforce the properties of rotation matrices, namely orthonormality. We therefore wish to estimate a proper rotation matrix R, which is closest to the estimated $\bar{\mathsf{R}}$ in the sense of the Frobenius matrix norm. This is done by computing the SVD of $\bar{\mathsf{R}}$:

$$\bar{\mathsf{R}} = \mathsf{U}\Sigma\mathsf{V}^{\mathsf{T}}$$

and computing, after noting that the singular values of a rotation matrix are all equal to 1:

$$\begin{pmatrix} \\ \mathsf{R} \\ \\ \end{pmatrix} = \begin{pmatrix} \\ \mathsf{U} \\ \\ \end{pmatrix} \begin{pmatrix} 1 & & \\ & 1 & \\ & & 1 \end{pmatrix} \begin{pmatrix} \\ \mathsf{V}^{\mathsf{T}} \\ \\ \end{pmatrix} = \mathsf{U}\mathsf{V}^{\mathsf{T}}$$

The final step is to verify that R has a determinant of $+1$, and if not to multiply it by -1.

Plane fitting

Assume we want to fit a plane to n 3D points $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$. The best fit is defined as the plane that minimizes the sum of squared orthogonal distances from the points to the plane. We represent a plane by a point $\mathbf{c}$ and a normal $\mathbf{n}$ (of unit length), so the distance from a point $\mathbf{x}_i$ to the plane is

$$(\mathbf{x}_i - \mathbf{c}).\mathbf{n}$$

and the best fit plane should therefore minimize

$$\sum_{i=1}^{n} ((\mathbf{x}_i - \mathbf{c}).\mathbf{n})^2$$

We first solve for $\mathbf{c}$ as the centroid of the points

$$\mathbf{c} = \frac{\sum_{i=1}^{n} \mathbf{x}_i}{n}$$

and form a $n \times 3$ matrix A

$$\mathsf{A}_{n\times 3} = \begin{pmatrix} \mathbf{p}_1 - \mathbf{c} \\ \mathbf{p}_2 - \mathbf{c} \\ \vdots \\ \mathbf{p}_2 - \mathbf{c} \end{pmatrix}$$

which allows to write the minimization as

$$\min_{\|\mathbf{n}\|=1} \left\|\mathsf{A}\mathbf{n}\right\|^2$$

After computing the SVD of $\mathsf{A}^\mathsf{T}\mathsf{A} = \mathsf{U}\Sigma\mathsf{V}^\mathsf{T}$ the solution for $\mathbf{n}$ is the column of V that corresponds to the smallest singular value. Note that the other two columns of V therefore represent the directions spanned by the fitted plane, and more precisely the column of U corresponding to the largest singular value corresponds to the direction of the best fitted line to the set of points.

Bibliography

[AW87] John Amanatides and Andrew Woo. A fast voxel traversal algorithm for ray tracing. In *Eurographics '87*, pages 3–10. Elsevier Science Publishers, Amsterdam, North-Holland, 1987.

[BB88] Aandrew Blake and Gavin Brelstaff. Geometry from specularities. In *Second International Conference on Computer Vision (Tampa,, FL)*, pages 394–403, Washington, DC,, 12 1988. Computer Society Press.

[BB90] Andrew Blake and Heinrich H. Bülthoff. Does the brain know the physics of specular reflection? *Nature*, 343(6254):165–168, 1990.

[Bla85] Andrew Blake. Specular stereo. In *IJCAI*, pages 973–976, 1985.

[BOIK01] Mitsuru Baba, Kozo Ohtani, Makoto Imai, and Tadataka Konishi. New laser rangefinder for three-dimensional shape measurement of specular objects. *Optical Engineering*, 40(1):53–60, January 2001.

[BS03] Thomas Bonfort and Peter Sturm. Voxel carving for specular surfaces. In *International Conference on Computer Vision*, pages 591–596, 2003.

[BSG06] Thomas Bonfort, Peter Sturm, and Pau Gargallo. General specular surface triangulation. In *Proceedings of the Asian Conference on Computer Vision, Hyderabad, India*, volume II, pages 872–881, jan 2006.

[CG00] Roberto Cipolla and Peter Giblin. *Visual motion of curves and surfaces*. Cambridge University Press, New York, NY, USA, 2000.

[CI01] Yaron Caspi and Michal Irani. Alignment of non-overlapping sequences. In *Internation Conference on Computer Vision*, pages 76–83, 2001.

[CT82] Robert L. Cook and Kenneth E. Torrance. A reflectance model for computer graphics. *ACM Trans. Graph.*, 1(1):7–24, 1982.

[FC02] Vincent Frémont and Ryad Chellali. Direct camera calibration using two concentric circles from a single view. In *International Conference on Artificial Reality and Telexistence*, pages 93–98, Tokyo, Japan, dec 2002.

[FTA04] Roland W. Flemming, Antonio Torralba, and Edward H. Adelson. Specular reflections and the perception of shape. *Journal of Vision*, 9(4):798–820, 2004.

[GN01] Michael Grossberg and Shree K. Nayar. A general imaging model and a method for finding its parameters. In *International Conference on Computer Vision*, pages 108–115, 2001.

[Gra53] Frank Gray. Pulse code communication. *U.S. patent*, (2 632 058), March, 17 1953.

[HBKM96] Mark Halstead, Brian Barsky, Stanley Klein, and Robert Mandell. Reconstructing curved surfaces from specular reflection patterns using spline surface fitting of normals. In *SIGGRAPH 96 Conference Proceedings*, pages 335–342, 1996.

[HD95] Radu Horaud and F. Dornaika. Hand-eye calibration. *International Journal of Robotics Research*, 14(3):195–210, jun 1995.

[HS88] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, pages 147–151, 1988.

[HTFG91] Xiao D. He, Kenneth E. Torrance, cois X. Sillion Fran and Donald P. Greenberg. A comprehensive physical model for light reflection. *SIGGRAPH Comput. Graph.*, 25(4):175–186, 1991.

[HZ00] Richard I. Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.

[KKH04] Markus C. Knauer, Jürgen Kaminski, and Gerd Hausler. Phase measuring deflectometry: a new approach to measure specular free-form surfaces. *Optical Metrology in Production Engineering*, 5457(1):366–376, 2004.

[KLKH05] Jürgen Kaminski, Svenja Lowitzsch, Markus Knauer, and Gerd Häusler. Full-field shape measurement of specular surfaces. In *Fringe 2005, The 5th International Workshop on Automatic Processing of Fringe Patterns*, pages 372–379. Wolfgang Osten (Ed.), Springer, Berlin, Heidelberg, New York, 2005.

[KS00] Kirios N. Kutulakos and Steven M. Seitz. A theory of shape by space carving. *IJCV*, 38(3):199–218, July 2000.

[KS05] Kiriakos N. Kutulakos and Eron Steger. A theory of refractive and specular 3d shape by light-path triangulation. In *Proc. 10th Int. Conf. Computer Vision*, 2005.

[KvD89] Jan J. Koenderink and Andrea J. van Doorn. *Photometric invariants related to solid shape*. MIT Press, Cambridge, MA, USA, 1989.

[MKZB01] Sebastian Magda, David J. Kriegman, Todd Zickler, and Peter N. Belhumeur. Beyond lambert: Reconstructing surfaces with arbitrary brdfs. In *Proc. 8th IEEE International Conference on Computer Vision*, volume II, pages 391–398, june 2001.

[MN05] Nigel J.W. Morris and Kiriakos N.Kutulakos. Dynamic refraction stereo. In *International Conference on Computer Vision*, volume 2, pages 1573–1580, 2005.

[MNY92] Paul M.Griffin, Lakshmi S. Narasimhan, and Soung R. Yee. Generation of uniquely encoded light patterns for range data acquisition. *Pattern Recognition*, 6:609–616, 1992.

[Nay88] Shree K. Nayar. Sphereo: Determining depth using two specular spheres and a single camera. In *1988 Cambridge Symposium on Advances in Intelligent Robotics Systems*, 1988.

[NSWS90] Shree K. Nayar, Arthur C. Sanderson, Lee E. Weiss, and DavidA. Simon. Specular surface inspection using structured highlight and gaussian images. *IEEE Transactions on Roboticsand Automation*, 6(2):208–218, 1990.

[NTO04] J. Farley Norman, James T. Todd, and Guy A. Orban. Perception of three-dimensional shape from specular highlights, deformations of shading, and other types of visual information. *Psychological Science*, 15:565–570, 2004.

[Oko76] Takanori Okoshi. *Three-Dimensional Imaging Techniques*. Academic Press, New York, 1976.

[ON95a] Michael Oren and Shree K. Nayar. Generalization of the lambertian model and implications for machine vision. *IJCV*, 14(3):227–251, April 1995.

[ON95b] Michael Oren and Shree K. Nayar. A theory of specular surface geometry. In *International Conference on Computer Vision*, pages 740–747, 1995.

[PC96] Won S. Park and Hyung S. Cho. Measurement of the 3-dimensional shapes of specular objects using recursive triangulation. In *Proc. of Pacific Conference on Manufacturing*, volume 1, pages 389–394, 1996.

[PFTV92] William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, Cambridge (UK) and New York, 2nd edition, 1992.

[Pho75] B. T. Phong. Illumination for computer generated pictures. *Commun. of the ACM*, 18(6), Jun 1975.

[Pol00] Marc Pollefeys. Tutorial on 3d modelling from images. `http://www.esat.kuleuven.ac.be/pollefey/tutorial/`, 2000.

[Pov96] Povray. The persistence of vision ray tracer. `www.povray.org`, 1996.

[RC01] Stefan Rahmann and Nikos Canterakis. Reconstruction of specular surfaces using polarization imaging. In *Computer Vision and Pattern Recognition*, volume 1, pages 149–155, 2001.

[SB06] Peter Sturm and Thomas Bonfort. How to compute the pose of an object without a direct view? In *Proceedings of the Asian Conference on Computer Vision, Hyderabad, India*, volume II, pages 21–31, jan 2006.

[Sch96] Howard Schultz. Shape reconstruction from multiple images of the ocean surface. *Photogrammetric Engineering and Remote Sensing*, 62(1):93–99, 1996.

[SCP04] Silvio Savarese, Min Chen, and Pietro Perona. Recovering local shape of a mirror surface from reflection of a regular grid. In *European Conference on Computer Vision*, 2004.

[SD97] Steve M. Seitz and Charles R. Dyer. Photorealistic scene reconstruction by voxel coloring. In *CVPR97*, pages 1067–1073, 1997.

[SFFP04] Silvio Savarese, Li Fei-Fei, and Pietro Perona. What do reflections tell us about the shape of a mirror. In *Proceedings of the First Symposium on Applied Perception in Graphics and Visualization*. ACM Press, 2004.

[SH04] Jan-Erik Solem and Anders Heyden. Estimating surface shape and extending known structure using specular reflections. In *International Conference on Pattern Recognition, Cambridge, UK*, 2004.

[SM99] Peter Sturm and Stephen Maybank. On plane-based camera calibration: A general algorithm, singularities, applications. In *Computer vision and Pattern Recognition*, pages 432–437, 1999.

[SNG03] Rahul Swaminathan, Shree K. Nayar, and Michael D. Grossberg. Framework for designing catadioptric projection and imaging systems. In *Proceedings IEEE Conf. on Computer Vision PROCAMS*, October 2003.

[SP94] François Sillion and Claude Puech. *Radiosity and Global Illumination*. Morgan Kaufmann Publishers, San Francisco, 1994. ISBN 1-558.

[SP02] Silvio Savarese and Pietro Perona. Local analysis for 3d reconstruction of specular surfaces — part ii. *European Conference on Computer Vision*, 2002.

[SPB04] Joachim Salvi, Jordi Pages, and Joan Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827–849, April 2004.

[SR04] Peter Sturm and Srikumar Ramalingam. A generic concept for camera calibration. In *Proceedings of the European Conference on Computer Vision*, volume 2, pages 1–13. Springer, May 2004.

[SS03] Daniel Scharstein and Richard Szeliski. High-accuracy stereo depth maps using structured light. In *CVPR (1)*, pages 195–202. IEEE Computer Society, 2003.

[SsH04] Jan-Erik Solem, Henrik Aanæs, and Anders Heyden. A variational analysis of shape from specularities using sparse data. In *2nd Int. Symposium on 3D Data Processing, Visualization and Transmission, Thessaloniki, Greece*, 2004.

[Stu00] Peter Sturm. Algorithms for plane-based pose estimation. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina, USA*, pages 1010–1017, June 2000.

[SW00] Amnon Shashua and Lior Wolf. On the structure and properties of the quadrifocal tensor. In *European Conference on Computer Vision (ECCV)*, Dublin, Ireland, 2000.

[TLGS03] Marco Tarini, Hendrik P.A Lensch, Michael Goesele, and Hans-Peter Seidel. 3d acquisition of mirroring objects. In *Research Report MPI-I-2003-4-001*. Max-Planck-Institut für Informatik, 2003.

[Tsa86] Roger Y. Tsai. An efficient and accurate camera calibration technique for 3d machine vision. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 364–374, 1986.

[YPW03] Ruigang Yang, Marc Pollefeys, and Greg Welch. Dealing with textureless regions and specular highlights-a progressive space carving scheme using a novel photo-consistency measure. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, pages 576–584, Washington, DC, USA, 2003. IEEE Computer Society.

[ZGB89] Andrew Zisserman, Peter Giblin, and Andrew Blake. The information available to a moving observer from specularities. *Image and Vision Computing*, 7:38–42, 1989.

[Zha00] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1330–1334, 2000.

[ZM98] Jiang Yu Zheng and Akio Murata. Acquiring 3D object models from specular motion using circular lights illumination. In *Procedings of the Sixth International Conference on Computer Vision (ICCV-98)*, pages 1101–1108, 1998.