

Roland Schellhorn

Objekterkennung mit OpenCV

103 - Recent Advances in Multimedia Processing

Abstract

Die Analyse unterschiedlicher Objekterkennungsmethoden der Open Source Library OpenCV wird anhand eines konkreten Beispiels auf die Effizienz in Hinblick auf Performance und Genauigkeit evaluiert. Als Testbilder dienen Bilder der kulturhistorischen Kanonen der Festung Kufstein, die in unterschiedlichen Blickwinkeln angefertigt und partiell in Kontrast und Helligkeit bearbeitet wurden. Eine neuartige Software unterstützte anschließend die Suche nach einer performanten und präzisen Lösung, welche sich zur Objekterkennung nach kulturhistorischen Kanonen auf mobilen Endgeräten eignen würde.

Keywords:

AR, VR, Feature Detection, Description, Matching, OpenCV, Find-Object, Dense, Fast, GFTT, MSER, ORB, SIFT, Star, SURF, BRISK

1. Einleitung

Die Open Source Library OpenCV bietet eine Menge an unterschiedlicher Objekterkennungsmethoden. Wie sich diese Methoden in Hinblick auf Performance, Genauigkeit und Kompatibilität untereinander unterscheiden, kann nur durch verschiedene Analysen aufgezeigt werden. Eine neuartige Software ermöglicht ein einfaches Kombinieren und Testen der auf OpenCV verfügbaren Objekterkennungsalgorithmen.

2. State of the Art

Bekannte Methoden zur Objekterkennung wie SIFT (Lowe 2004) und SURF (Bay, Ess et al. 2008) existieren bereits seit dem Jahre 2004 bzw. 2006. Für SIFT erschien 2011 auch eine Affine Methode des Algorithmus, welche unter dem Namen ASIFT (Yu and Morel 2011) bekannt ist. Während SIFT und SURF auf eine textuelle Keypoint Beschreibung der Eigenschaften eines Bildes setzen, verwenden aktuelle Methoden wie ORB (Rublee, Rabaud et al. 2011), BRISK (Leutenegger, Chli et al. 2011) und BRIEF (Calonder et al. 2010) eine binäre Beschreibung. Dadurch verbessert sich einerseits die Performance, jedoch in Hinblick auf die Robustheit schneiden diese Algorithmen schlechter in den Testergebnissen ab. Eine ähnliche Arbeit nannte GLOH und SIFT als nahezu gleichauf schnellste und robusteste Lösung (Mikolajczyk and Schmid 2005). Im Vergleich zwischen SIFT, SURF und ORB erzielt SURF bis zu 50% und ORB bis zu 95% bessere Durchlaufzeiten, was jedoch Einbußen der Invarianz zur Folge hat (Dwarakanath, Eichhorn et al. 2012). Ein Vergleich von neun Feature Detektoren und Deskriptoren in 42 unterschiedlichen Kombinationen zeigt, dass SIFT mit SIFT oder

1

SURF als effizienteste Variante zugleich ORB als performanteste Lösung für unterschiedlichste Einsatz Möglichkeiten gesehen werden kann (Mukherjee et al. 2015).

3. Testumgebung

Um die verfügbaren Objekterkennungsmethoden der OpenCV Library zu testen, zu kombinieren und anzupassen, wird ein Programm namens Find-Object verwendet. Das Programm läuft auf einem Windows 7 Rechner mit einem Intel Core i7 950 3.07GHz Prozessor und 6GB RAM Arbeitsspeicher. Die Versionsnummer der verwendeten Software lautet 0.5.1. Die Qualität der verwendeten Bilder befindet sich zwischen 780p und 1080p.

Die nachfolgende Abb. 1 zeigt einen Screenshot des Programmes. Zur Visualisierung des Suchobjektes und dessen Features und Matches dient Feld 1. Das Programm erlaubt einen Input als Source oder via Webcam. Feld 2 beinhaltet alle Objekte/Bilder, die als Vergleich für Matches verwendet werden. Verschiedene Einstellungen zu den jeweilig verwendeten Algorithmen kann in Feld 3 vorgenommen werden.

Feld 4 visualisiert die Likelihood Wahrscheinlichkeits-Funktion. Alle zum Programm relevanten Statistiken werden in Feld 5 dargestellt. Abschließend stellt Bereich 6 bekannte Tab Einstellungen dar.

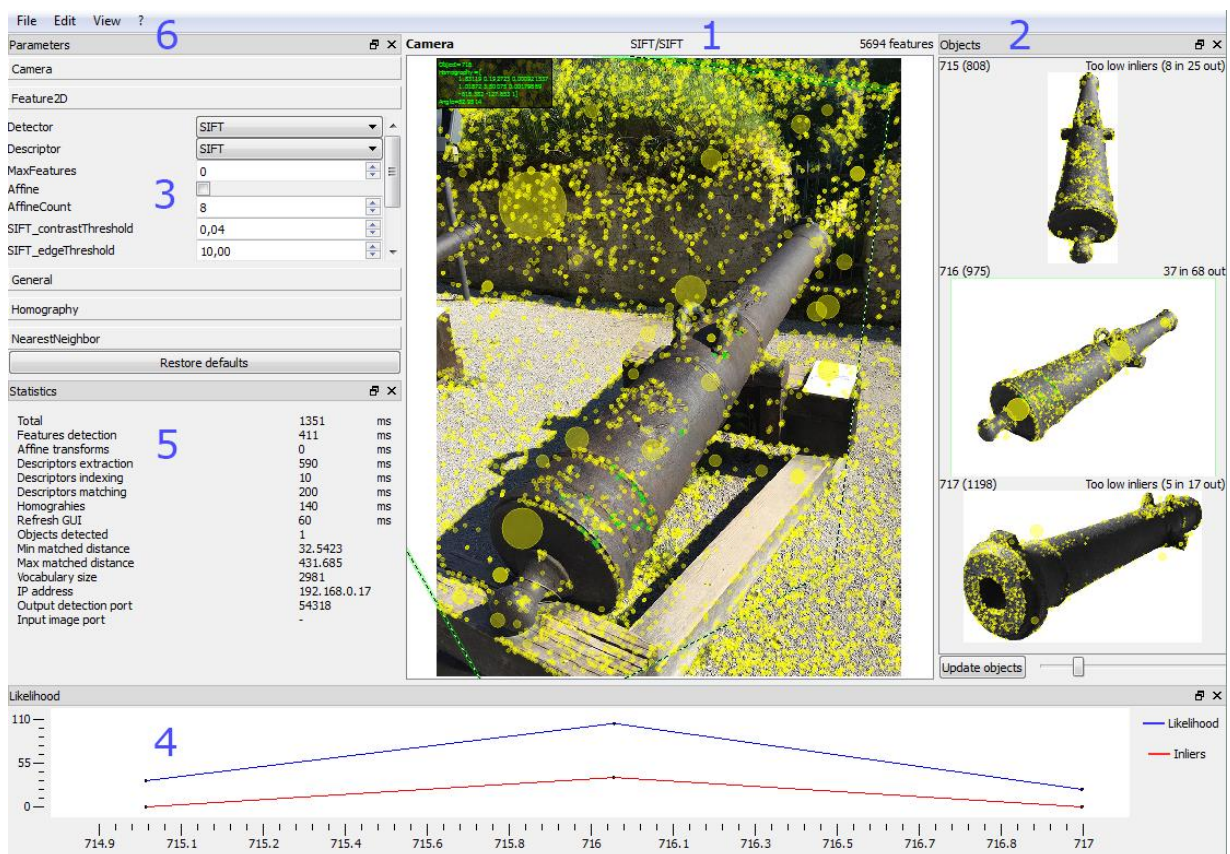


Abb. 1: Programm Screenshot Übersicht (Eigene Darstellung)

4. Objekteigenschaften

Objekte unterscheiden sich mehr oder weniger ausgeprägt von anderen Objekten. Demzufolge bedarf das Erkennen von Eigenschaften je nach Objekt oft unterschiedlicher Methoden. Für Methoden, die visuelle Vergleiche darstellen, liegen Eigenschaften in Form und Oberflächenbeschaffenheit wie Farbe und Struktur im Vordergrund. Diese Methoden können durch weitere Methoden und Sensoren unterstützt werden, die zusätzlich beispielsweise Temperatur und Bewegung beschreiben.

Als Schlussfolgerung können folgende Aussagen getroffen werden:

- Je mehr Eigenschaften aus einem Objekt heraus bezogen werden können, umso leichter und genauer lässt es sich bestimmen.
- Je deutlicher sich ein Objekt aus seiner Umgebung heraus absetzt und sich von anderen Objekten unterscheidet, desto leichter ist es auch, ein solches zu erkennen.

Eine für das Testverfahren verwendete kulturhistorische Kanone weist folgende Eigenschaften auf:

Der Großteil aller Kanonen ist länglich, rund und hat eine lochartige Öffnung. Des Weiteren besitzen Kanonen Zündvorrichtungen hinten. Sie werden von vorne nach hinten breiter und zusätzlich haben sie eine Vorrichtung, um einerseits auf Wagen oder Stationen platziert werden zu können und andererseits, um die Flugbahn zu beeinflussen. Abb. 2 zeigt eine im Testverfahren verwendete Verteidigungskanone mit ihren Merkmalen und Beispiele anderer Kanonenarten.

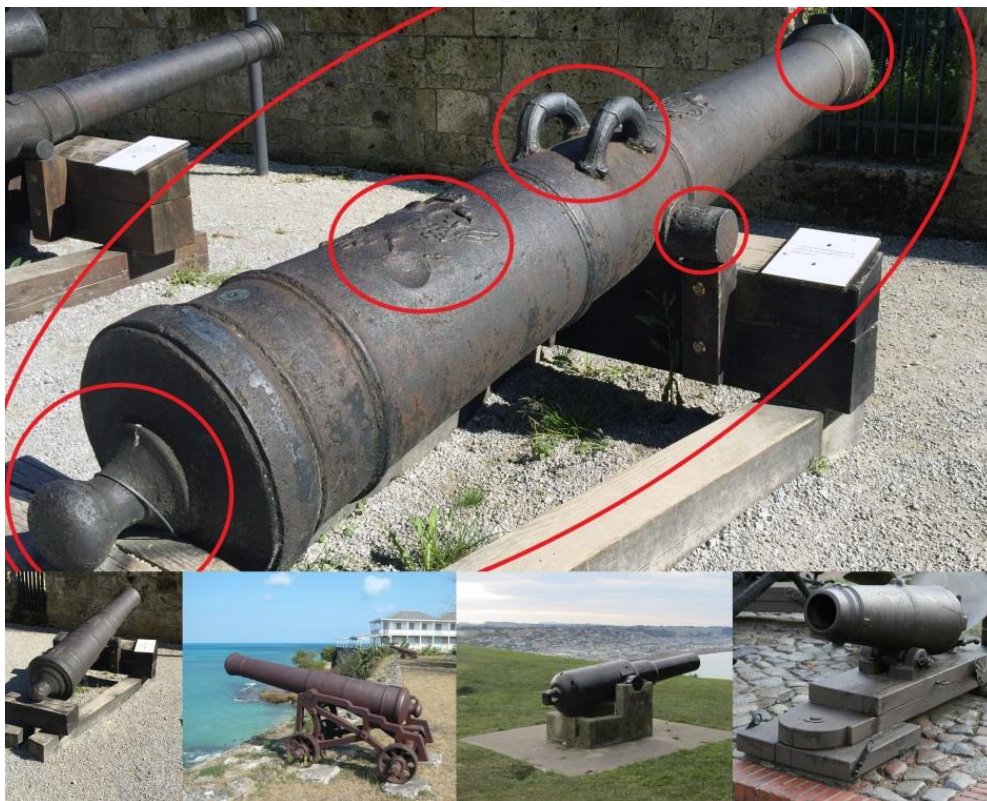


Abb. 2: Merkmale Suchobjekt (Eigene Darstellung, Bilder Flickr)

5. Testergebnisse

Die Testreihe enthielt 446 Bilder von kulturhistorischen Kanonen, davon sind 96 Bilder der Festung Kufstein, welche in verschiedenen Blickwinkeln aufgenommen und teilweise in Kontrast, Helligkeit und Schärfe überarbeitet wurden, um verschiedene Tageszeiten sowie Lichtverhältnisse zu simulieren. Untersucht wurden alle möglichen Kombinationen nachfolgender Detektoren und Deskriptoren: Dense, Fast, Star, GFTT, MSER, Brief, ORB, BRISK, FREAK, SIFT, SURF. Abb. 3 zeigt eine Testgruppe veränderter kulturhistorischer Bilder.



Abb. 3: Verändertes Original (Eigene Darstellung)

Die nachfolgende Abb. 4 illustriert das Ergebnis der Evaluierung unterschiedlicher Detektoren in Bezug auf die Menge der extrahierten Keypoints und die dafür aufgebrauchte Zeit. Gelbe Punkte visualisieren die extrahierten Keypoints. Die erste Zahl gekennzeichnet mit einem F für Features stellt die Anzahl der Keypoints dar. Die zweite Zahl veranschaulicht die dafür aufgebrauchte Zeit in Millisekunden.

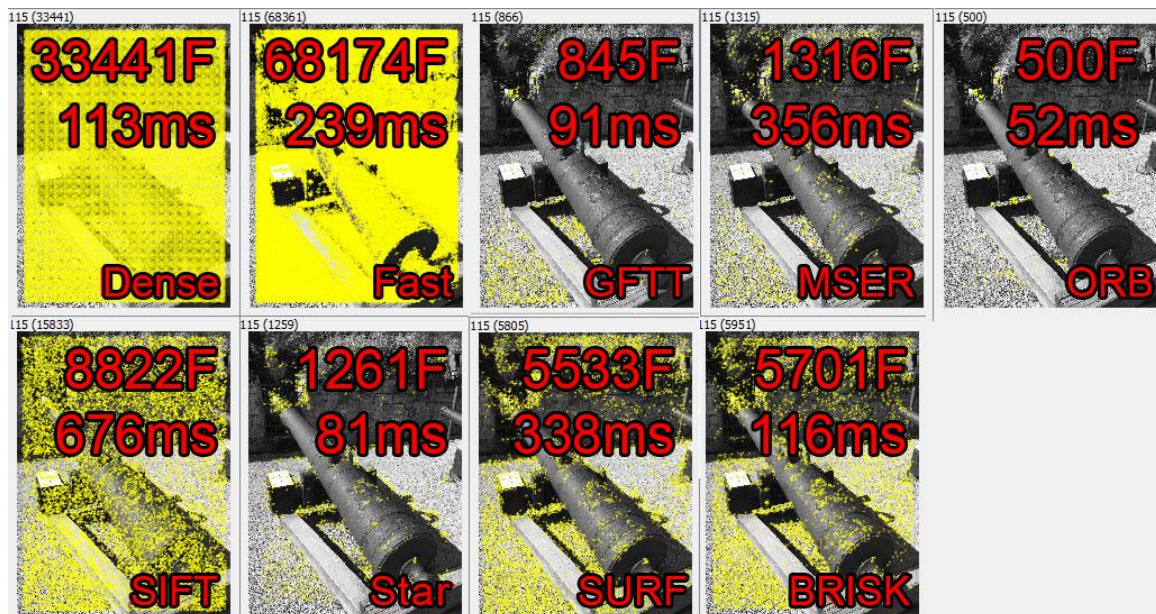


Abb. 4: Detected Features (Eigene Darstellung)

Um die verfügbaren möglichen Detektor- und Deskriptor-Verfahren auf ihre Effizienz hin zu untersuchen, wurden zwei selbige Kanonen in unterschiedlich aufgenommenen Blickwinkeln gegenübergestellt. Die verschiedenen Kombinationen der Algorithmen konnten das Bild größtenteils wiedererkennen. Die ausschlaggebenden Bereiche, welche zur eindeutigen Erkennung herangezogen wurden, befanden sich jedoch oftmals außerhalb des gesuchten Objektes im Hintergrund, da sich dort der markanteste Bereich und somit auch die stärksten Features befanden. Die rote Kreismarkierung, siehe Abb. 4, stellt die für das Match wiedererkannten Features als grünen Punkt illustriert dar.



Abb 4: Erkannte Matches im Hintergrund (Eigene Darstellung)

Um zu verhindern, dass Matches zwischen Keypoints außerhalb des gesuchten Objektes entstehen, darf die Vergleichsseite nur Keypoints vom Objekt besitzen. Bei einem neuerlichen Durchlauf erzielte der Fast Detektor in Verbindung mit SIFT als Deskriptor die besten Ergebnisse. Die gefundenen Matches beliefen sich auf eine Zahl von 489 bei einer Durchlaufzeit von 1669ms, in Abb. 5 dargestellt.

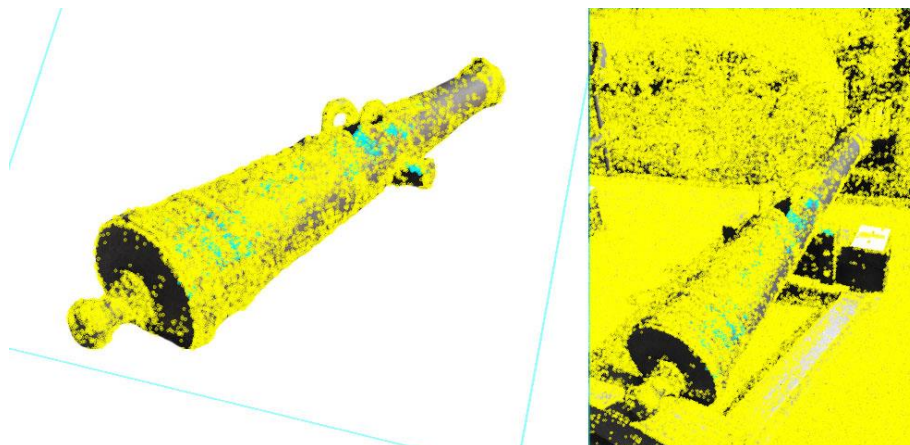


Abb 5: Fast / SURF mit Transparenter Schablone (Eigene Darstellung)

Zur Analyse der Robustheit, bezogen auf Skalierung und Rotation, wurden Bilder derselben Kanone in unterschiedlichen Blickwinkeln, Auflösungen, Helligkeiten und Kontrasten verwendet, die mit einer ausgeschnittenen Vorlage verglichen wurden. Abb. 6 veranschaulicht dieses Ergebnis. Ein erfolgreiches Match kann durch angehakte Kästchen identifiziert werden.

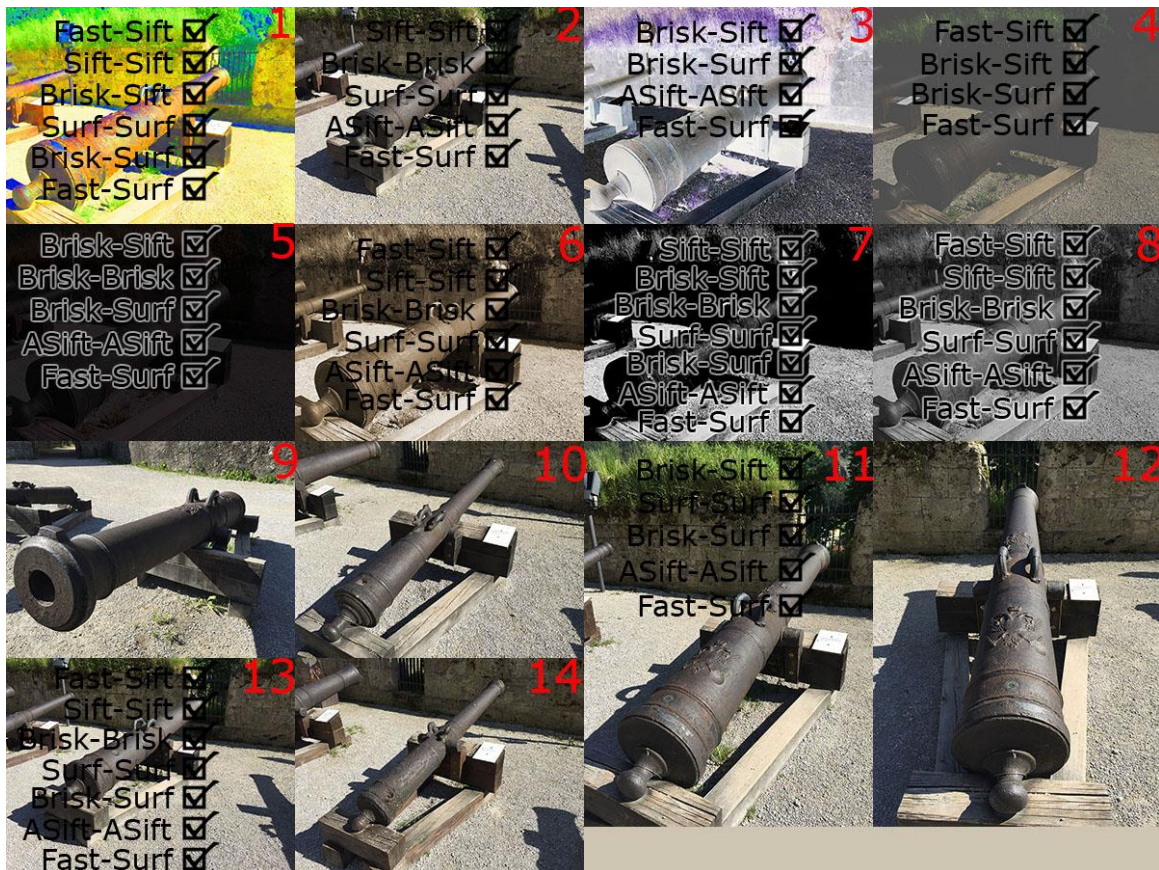


Abb 6: Erkannte Objekte im fünften Testverfahren (Eigene Darstellung)

6. Fazit und Zusammenfassung

Die Methode von FAST als Deskriptor in Verbindung mit SURF als Detektor ist besonders robust gegenüber geometrischen Transformationen, Beleuchtung und Noise. Aufgrund dieser Wiedererkennung von 10 aus 12 möglichen Bildern würde sich dieses Verfahren als Erkennung von kulturhistorischen Kanonen am besten eignen. Die Kombination aus dem BRISK Detektor und dem SURF Deskriptor stellte sich als das schnellste Verfahren heraus. Jedoch zeigte sich BRISK anfällig bei Rotationen und Größenänderungen. Zusätzlich ist die Menge der Features, welche zur eindeutigen Erkennung herangezogen werden, meist sehr gering. Um mit einem schnelleren Verfahren dieselben Ergebnisse zu erzielen, müsste im Gegenzug die Menge der Schablonen zur Erkennung erhöht werden. Nur durch weitere Tests kann festgestellt werden, ob das Verfahren dadurch weiterhin schneller als ein sehr robustes Verfahren ist. Aus der Testreihe mit 446 Bildern, wovon 350 Bilder kulturhistorischer Kanonen der Plattform Flickr entnommen wurden, wurde keine dieser Kanonen fälschlicherweise als Match identifiziert. Einerseits ist es hervorzuheben, dass keine ähnlichen Kanonen als dieselben erkannt wurden. Andererseits kann somit keine leicht veränderte Kanone derselben Baureihe erkannt werden, was zur damaligen Zeit der Handarbeit keine Seltenheit darstellte. Durch die leistungsstarke Entwicklung mobiler Endgeräte kann durchaus mit einer Anwendung diese Art in näherer Zukunft gerechnet werden.

Literaturliste/Quellenverzeichnis:

Alahi, Alexandre (2012): FREAK: Fast Retina Keypoint. In: Computer Vision and Pattern Recognition (CVPR). 510-517.

Rublee, Ethan (2011): ORB: An efficient alternative to SIFT or SURF. In: Computer Vision (ICCV). Barcelona: IEEE, pp 2564 - 2571.

Yu, Guoshen (2011) ASIFT: An Algorithm for Fully Affine Invariant Comparison, France: IPOL Journal Image Processing On Line.

Bay, Herbert (2006): SURF: Speeded Up Robust Features. In: Computer Vision – ECCV 2006. 404-417.

Itseez, (2015): OpenCV Open Source Computer Vision Library. www.opencv.org (9.5.2015)

Labbé, M. (2015): Find-Object. introlab.github.io/find-object/ (28.4.2015)

Lowe, D. G. (2004): Distinctive Image Features from Scale-Invariant Keypoints. In: International Journal of Computer Vision. 91-110.

Calonder, Michael (2010) BRIEF: Binary Robust Independent Elementary Features. In: Computer Vision – ECCV 2010. 778-792.

Leutenegger, Stefan (2011): BRISK: Binary Robust Invariant Scalable Keypoints. In: Computer Vision (ICCV). 2548-2555.

Krystian Mikolajczyk, Cordelia Schmid (2005): A Performance Evaluation of Local Descriptors.

Deepak Dwarakanath, Alexander Eichhorn, Pål Halvorsen, Carsten Griwodz (2012): Evaluating performance of feature extraction methods for practical 3D imaging systems. In: IVCNZ '12 Proceedings of the 27th Conference on Image and Vision Computing New Zealand, pp. Pages 250-255.

Dibyendu Mukherjee, Q. M. Jonathan Wu, Guanghui Wang, (2015): A comparative experimental study of image feature detectors and descriptors, In: Machine Vision and Applications May 2015, Volume 26, Issue 4, pp 443-466.