

abnt-nbr10520=2002

Tese apresentada à Pró-Reitoria de Pós-Graduação e Pesquisa do Instituto Tecnológico de Aeronáutica, como parte dos requisitos para obtenção do título de Mestre em Ciências no Curso de Engenharia Eletrônica e Computação, Área de Informática

Bruno Duarte Corrêa

**AVALIAÇÃO DE TÉCNICAS DE
RECONHECIMENTO DE PADRÕES EM
AMBIENTES AERONAUTICOS**

Tese aprovada em sua versão final pelos abaixo assinados:

Prof. Dr. Luís Gonzaga Trabasso
Orientador

Prof. Dr. Nelson José Issa de Macedo
Co-orientador

Prof. Dr. Celso Massaki Hirata
Pró-Reitor de Pós-Graduação e Pesquisa

Campo Montenegro
São José dos Campos, SP - Brasil

2015

Dados Internacionais de Catalogação-na-Publicação (CIP)

Divisão Biblioteca Central do ITA/CTA

Duarte Corrêa, Bruno

Avaliação de técnicas de reconhecimento de padrões em ambientes aeronauticos / Bruno Duarte Corrêa.

São José dos Campos, 2015.

70f.

Tese de Mestrado – Curso de Engenharia Eletrônica e Computação. Área de Informática – Instituto Tecnológico de Aeronáutica, 2015. Orientador: Prof. Dr. Luís Gonzaga Trabasso. Co-orientador: Prof. Dr. Nelson José Issa de Macedo.

1. Teses. 2. Estilos. 3. Italus. I. Centro Técnico Aeroespacial. Instituto Tecnológico de Aeronáutica. Divisão de Ciência da Computação. II. Título.

REFERÊNCIA BIBLIOGRÁFICA

DUARTE CORRÊA, Bruno. **Avaliação de técnicas de reconhecimento de padrões em ambientes aeronauticos**. 2015. 70f. Tese de Mestrado – Instituto Tecnológico de Aeronáutica, São José dos Campos.

CESSÃO DE DIREITOS

NOME DO AUTOR: Bruno Duarte Corrêa

TÍTULO DO TRABALHO: Avaliação de técnicas de reconhecimento de padrões em ambientes aeronauticos.

TIPO DO TRABALHO/ANO: Tese / 2015

É concedida ao Instituto Tecnológico de Aeronáutica permissão para reproduzir cópias desta tese e para emprestar ou vender cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte desta tese pode ser reproduzida sem a autorização do autor.

Bruno Duarte Corrêa

Av. ABC, 1000

CEP 12.000-000 – São José dos Campos–SP

AVALIAÇÃO DE TÉCNICAS DE RECONHECIMENTO DE PADRÕES EM AMBIENTES AERONAUTICOS

Bruno Duarte Corrêa

Composição da Banca Examinadora:

Prof. Dr.	Luís Gonzaga Trabasso	Presidente	-	ITA
Prof. Dr.	Luís Gonzaga Trabasso	Orientador	-	ITA
Prof. Dr.	Nelson José Issa de Macedo	Co-orientador	-	EMBRAER
Prof. Dr.	Ricardo Bedin França	Membro Externo	-	UYYY
Prof. Dr.	Emília	Membro	-	ITA

Aqui pode ser escrita uma dedicatória. Não é obrigatória.

Agradecimientos

"Persistence is the shortest path to success."

— CHARLES CHAPLIN

Resumo

O reconhecimento de objetos em uma cena para posterior uso em realidade aumentada depende de diversas variáveis, causando a necessidade do uso de diversas técnicas específicas para cada cenário, sendo portanto, um estudo de fronteiras para a melhor escolha do algoritmo de registro de acordo com a aplicação em questão de grande valia para o meio acadêmico. Esta tese se propõe a estudar, categorizar e traçar fronteiras das técnicas conhecidas tendo como caso de uso manutenção de aeronaves feita dentro de centros fechados.

Abstract

Recognition of objects in a scene for later use in augmented reality depends on several variables, causing the need to use several specific techniques for each scenario, and therefore a border study to the best choice of the registration algorithm according to application in question, of great value to academia. This thesis aims to examine, categorize and draw boundaries of the known techniques having as use case maintenance of aircraft made in closed centers.

Sumário

LISTA DE FIGURAS	x
LISTA DE ABREVIATURAS E SIGLAS	xii
1 MOTIVAÇÃO	13
2 OBJETIVO	15
3 CONTEXTO	16
3.1 Variáveis de contorno	16
3.2 Cenário	17
4 ESTRUTURA DO TEXTO	18
5 CONCEITOS	20
5.1 Realidade Aumentada	20
5.2 Head-Mounted Displays	20
5.2.1 Feed-Through	21
5.2.2 See-Through	22
5.2.3 Projetores	22
5.2.4 Monitores	22
5.3 Modos de Aquisição	23

5.3.1	Multiview	23
5.3.2	Multitemporal	23
5.3.3	Multimodal	24
5.3.4	Cena para o Modelo	24
5.4	Características Locais	24
5.4.1	Propriedades da Característica Local ideal	24
5.4.2	Features	25
5.5	Cadência	27
5.6	Camera	28
5.6.1	Distorção Radial	28
5.6.2	Distorção Tangencial	29
5.7	Modelo de Câmera	30
5.8	Reconhecimento	31
5.8.1	Detecção de características	32
5.8.2	Correspondência de características	33
5.9	Algoritmos de Reconhecimento	33
5.9.1	BRISK	33
5.9.2	FAST	34
5.9.3	FREAK	35
5.9.4	MSER	35
5.9.5	ORB	35
5.9.6	SIFT - Scale-Invariant Feature Transform	35
5.9.7	STAR	38
5.9.8	SURF - Speeded Up Robust Feature	38
6	METODOLOGIA	40

SUMÁRIO	xi
6.1 Definição de parâmetros	40
6.2 Análise de dados	40
7 PROTÓTIPOS	42
7.1 Testes de reconhecimento de padrões	42
7.2 Ambiente	44
7.2.1 OpenCV	44
8 RESULTADOS	45
8.1 Análise de parâmetros pontuais	45
8.1.1 BRISK	46
8.1.2 FAST	48
8.1.3 FREAK	49
8.1.4 MSER	51
8.1.5 ORB	53
8.1.6 SIFT	55
8.1.7 STAR	57
8.1.8 SURF	59
8.2 Análise de resultados	61
8.3 Homografia	63
9 CONCLUSÃO	66
10 TRABALHOS FUTUROS	67
REFERÊNCIAS BIBLIOGRÁFICAS	69
GLOSSÁRIO	71

Lista de Figuras

FIGURA 3.1 – Posicionamento da LRU	17
FIGURA 5.1 – Arquitetura do Feed-Trough.	21
FIGURA 5.2 – Arquitetura do See-Trough.	22
FIGURA 5.3 – Realidade Aumentada com projetores.	23
FIGURA 5.4 – Reconhecimento de padrões.	26
FIGURA 5.5 – Regiões de reconhecimento de padrões.	26
FIGURA 5.6 – Distorção radial.	29
FIGURA 5.7 – Componentes de Distorção radial.	29
FIGURA 5.8 – Distorção tangencial.	30
FIGURA 5.9 – Componentes de Distorção tangencial.	30
FIGURA 5.10 – Camera no espaço vetorial.	31
FIGURA 5.11 – Ilustração do procedimento de reconhecimento com features locais.	31
FIGURA 5.12 – FAST	34
FIGURA 5.13 – SIFT	35
FIGURA 5.14 – Difference of Gaussian	36
FIGURA 5.15 – Difference of Gaussian	37
FIGURA 5.16 – Box Filtering	39
FIGURA 8.1 – Resultado de testes individuais de performance com BRISK	46

FIGURA 8.2 – Resultado de testes individuais de tempo com BRISK	47
FIGURA 8.3 – Resultado de testes individuais de preformance com FAST	48
FIGURA 8.4 – Resultado de testes individuais de tempo com FAST	49
FIGURA 8.5 – Resultado de testes individuais de preformance com FREAK	50
FIGURA 8.6 – Resultado de testes individuais de tempo com FREAK	51
FIGURA 8.7 – Resultado de testes individuais de preformance com MSER	52
FIGURA 8.8 – Resultado de testes individuais de tempo com MSER	53
FIGURA 8.9 – Resultado de testes individuais de preformance com ORB	54
FIGURA 8.10 – Resultado de testes individuais de tempo com ORB	55
FIGURA 8.11 – Resultado de testes individuais de preformance com SIFT	56
FIGURA 8.12 – Resultado de testes individuais de tempo com SIFT	57
FIGURA 8.13 – Resultado de testes individuais de preformance com STAR	58
FIGURA 8.14 – Resultado de testes individuais de tempo com STAR	59
FIGURA 8.15 – Resultado de testes individuais de preformance com SURF	60
FIGURA 8.16 – Resultado de testes individuais de tempo com SURF	61
FIGURA 8.17 – Performance quanto à variações de escala, iluminação, rotação e gaussian blur	62
FIGURA 8.18 – Performance quanto à variações de escala, iluminação, rotação e gaussian blur sem os algoritmos SIFT e SURF	62
FIGURA 8.19 – Homografia com variação variação de Blur.	63
FIGURA 8.20 – Homografia com variação variação de iluminação.	64
FIGURA 8.21 – Homografia com variação variação de Rotação.	64
FIGURA 8.22 – Homografia com variação variação de Escala.	65

Lista de Abreviaturas e Siglas

HMD - Head-Mounted Display

AR - Augmented Reality

VR - Virtual Reality

DoG - Difference of Gaucians

LoG - Laplacian of Gaucians

1 Motivação

O reconhecimento de estruturas e sistemas de forma automática no campo da manutenção auxilia além de garantir maior confiabilidade no diagnóstico de problemas pode propiciar uma ferramenta de capacitação entre várias possibilidades. Um dos mais básicos problemas atualmente limitando o ramo da Realidade Aumentada é a etapa de registro. A Realidade Aumentada prevê imersão entre o mundo virtual e o mundo real e por isso para que a experiência de imersão seja coerente é necessário que os dois mundos estejam bem sincronizados e propriamente alinhados. Para algumas aplicações tal sincronia aumenta a experiência, entretanto existem aplicações que tal alinhamento é primordial, por exemplo em aplicações médicas por exemplo em uma aplicação de biopsia. Se o objeto não estiver no espaço e tempo da realidade, a informação fornecida ao cirurgião poderá por em risco a vida do paciente. Na maioria das aplicações de tempo real, problemas de registro podem invalidar o uso da Realidade Aumentada. Um outro problema que pode ocorrer com falhas de registro é acentuado por um fenômeno conhecido como visual capture [Welch78] que é a tendência do cérebro acreditar mais no que vê do que no que sente, ouve, etc. Nesses casos o sentido da visão tende a sobrepor os outros sentidos. Assim como um ventríloco consegue enganar quem assiste um show acreditando que o som sai da boca do boneco o usuário de uma aplicação de realidade aumentada tenderá a acreditar no que vê, mesmo que esteja defasado no espaço/tempo. No caso do erro se tornar sistemático o usuário

tende a se acostumar inconscientemente e se adaptar ao erro, corrigindo o efeito. Erros de registro são difíceis de controlar adequadamente devido à grande precisão requerida das diversas fontes de erro. As fontes de erro podem ser divididas em estáticas e dinâmicas sendo as estáticas contornadas com calibração prévia de sensores entretanto os erros dinâmicos são mais difíceis porque são susceptíveis a tempo diferença de tempo entre o real e o apresentado na tela e com o acúmulo de erro. O reconhecimento de objetos na cena permeia também:

- O contexto da cena, sendo que com conhecimento prévio do cenário se torna bem mais fácil;
- O material do qual o objeto é feito, porque caso seja feito de materiais reflexivos, os algoritmos podem confundir o reflexo de outros objetos com informações a reconhecer;
- O tamanho do objeto, pois de acordo com a escala do objeto, muitas informações que poderiam ser boas para o reconhecimento podem estar próximas demais dificultando o posterior casamento de informações. . . .

Portanto para que as diversas fontes de erros dinâmicos não sejam um impeditivo para o reconhecimento, de acordo com a cena, algoritmos diferentes devem ser selecionados por terem peculiaridades e características que garantam um registro direcionados ao tipo de desafio que encontrarão, além de já ter informações prévias, o que facilita na seleção de características.

2 Objetivo

O presente trabalho tem como objetivo geral avaliar o ambiente de manutenção aeronáutico interno, no contexto de janelas de inspeção e traçar estratégias de reconhecimento de itens de manutenção. Para a consecução do objetivo geral, foram definidos os seguintes objetivos específicos:

- Avaliar as algoritmos clássicos de reconhecimento;
- Levantar características padrão dos objetos no contexto;
- Avaliar limites de percepção humanas das características padrão;
- Avaliar algoritmo mais adequado para o contexto.

A proposta dessa tese é a partir do cenário de manutenção de aeronaves e da proposta de utilização de realidade aumentada, determinar a melhor estratégia de reconhecimento de peças para que posteriormente seja utilizado em ferramentas de auxílio na manutenção por meio da realidade aumentada

3 Contexto

O contexto dessa tese prevê o cenário de manutenção com o uso de realidade aumentada como uma ferramenta para auxiliar nas tarefas rotineiras portanto algumas variáveis devem ser consideradas para garantir a viabilidade de implantação da abordagem:

- Velocidade de reconhecimento
- Qualidade do reconhecimento
- Invariância quanto à parâmetros ambientais

3.1 Variáveis de contorno

O cenário de reconhecimento de objetos dentro da aeronave traz alguns desafios que devem ser contornados

- Pouca iluminação em ambientes internos
- Objetos muito parecidos entre si
- Alguns objetos com textura
- Objeto brilhante

3.2 Cenário

O uso da realidade aumentada em manutenção de aeronaves pode trazer ganho no que tange fornecer informações de procedimentos ao mecânico ou mesmo previsão de falhas ou reconhecimento de regiões com falha.

Como caso de uso será adotado a janela de inspeção frontal como mostrado na imagem 3.1 que mostra onde fica localizado na aeronave Embraer 190.



FIGURA 3.1 – Posicionamento da LRU

Foi selecionado um objeto sem texturas e com material brilhante, na janela de inspeção por sofrer mais influência em variação de iluminação e ser único na janela de inspeção em questão.

4 Estrutura do texto

Este trabalho fundamenta-se em 9 capítulos, conforme descritos abaixo:

- Capítulo 1 apresenta a motivação do presente trabalho, levantando as necessidades e limitações impostas pelo ambiente e o fato de haverem diversas técnicas de reconhecimento de características e a necessidade de selecionar a adequada para o contexto;
- Capítulo 2 descreve o escopo do trabalho;
- Capítulo 3 descreve o contexto em que serão feitos os testes e análises bem como o cenário em que as amostras serão retiradas;
- Capítulo 4 tem informações suficientes para o entendimento das análises, descrevendo conceitos básicos e as técnicas de reconhecimento que foram comparadas;
- Capítulo 5 descreve a metodologia de análise adotada;
- Capítulo 6 descreve os protótipos desenvolvidos para a análise;
- Capítulo 7 descreve os resultados obtidos comparando-se as técnicas;

- Capítulo 8 conclui a partir dos dados apresentados no capítulo de resultados qual a técnica mais adequada para reconhecimento de características para o caso de uso descrito;
- Capítulo 9 descreve possíveis trabalhos futuros.

5 Conceitos

5.1 Realidade Aumentada

A realidade aumentada como citado em ([AZUMA, 1997](#)) é uma técnica de visão computacional em que valendo-se de artefatos do mundo real tem por objetivo causar sensação de imersão do usuário em um ambiente aumentado por artefatos virtuais, ao contrário de ambientes puramente virtuais como é comum em aplicações de realidade virtual. Idealmente o mundo virtual se torna imersivo o suficiente para que o usuário não consiga distinguir o real do virtual. Alguns autores definem AR como tendo a necessidade de utilizar-se interfaces visuais portáteis para que a usabilidade tenha mais coerência com a proposta inicial de garantir uma experiência imersiva. As imagens são obtidas por câmeras e o resultado apresentado em dispositivos como projetores ou displays como monitores, tablets ou head-mounted display (HMD). Como mostrado em ([SILVA; OLIVEIRA; GIRALDI, 2003](#))

5.2 Head-Mounted Displays

É um equipamento utilizado na cabeça de forma que as duas mãos do usuário fiquem livres e tem por objetivo exibir imagens e áudio, sendo uma interface muito utilizada tanto

em RV quanto em RA. Os HMD basicamente são dispositivos constituídos de duas telas posicionadas frente ao olho do usuário. Com duas telas, a tecnologia pode ser empregada para exibir imagens estereoscópicas apresentando os respectivos pontos de vista de cada olho para cada tela, o que contribui em muito na experiência de imersão. Os HMDs funcionam também como dispositivos de entrada de dados, porque contém sensores de rastreamento que medem a posição e orientação da cabeça, transmitindo esses dados ao computador. Existem dois tipos de HMDs: Feed-Through e See-Through

5.2.1 Feed-Through

São dispositivos que é um sistema fechado de visualização de imagens, em que o usuário consegue enxergar somente o que é mostrado no display, sendo assim o resultado apresentado é sempre a soma da imagem real com informações superpostas

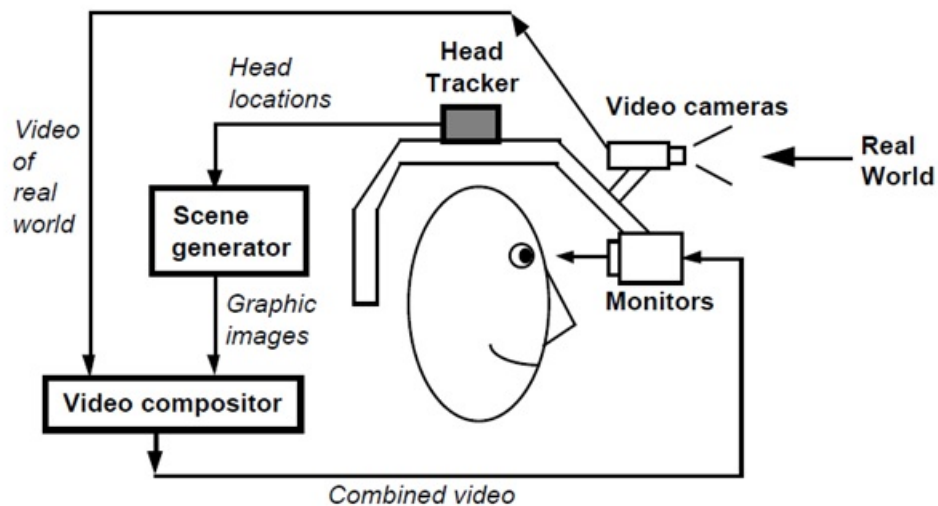


FIGURA 5.1 – Arquitetura do Feed-Trough.

5.2.2 See-Through

São dispositivos construídos com lentes translúcidas em que o usuário enxerga o mundo real e com algum tipo de sistema que sobreponha na lente as informações adicionais

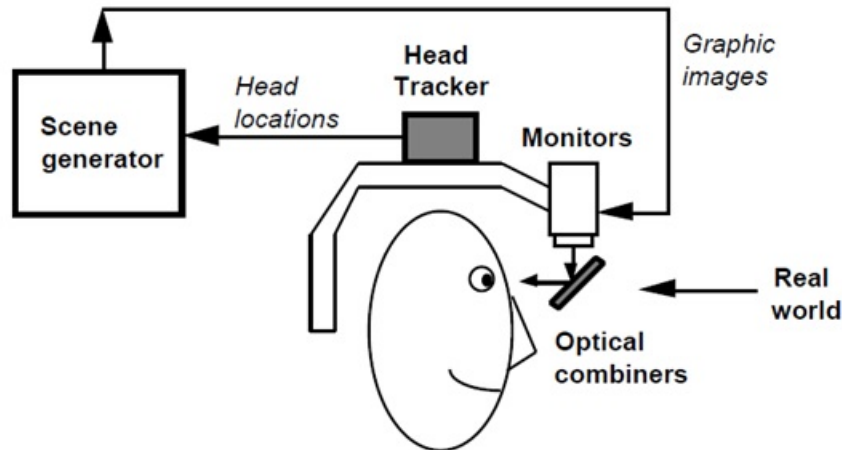


FIGURA 5.2 – Arquitetura do See-Trough.

5.2.3 Projetores

O uso de projetores possibilita uma abordagem de realidade aumentada diferente porque pode ser utilizada para cobrir superfícies largas, projetando sobre objetos como carros, pessoas, prédios, etc. Um problema dessa abordagem é que a calibração se faz necessária em várias situações.

5.2.4 Monitores

O uso de monitores reduz bastante o custo da aplicação apesar de ter perda de imersão por ser um método de visualização indireta, o que implica o usuário ficar olhando na direção do monitor, entretanto existe a possibilidade de compartilhar os resultados da RA

com mais de uma pessoa ao mesmo tempo.



Fonte: (IMMERSSION, 2014)

FIGURA 5.3 – Realidade Aumentada com projetores.

5.3 Modos de Aquisição

As imagens utilizadas como entrada para os algoritmos podem ser adquiridas de diversas formas.

5.3.1 Multiview

Imagens da mesma cena capturadas de pontos de vista diferentes

5.3.2 Multitemporal

Imagens da mesma cena capturados em momentos diferentes.

5.3.3 Multimodal

Imagens da mesma cena capturados por sensores diferentes para integrar informações de diferentes fontes e assim ter uma representação mais detalhada do ambiente.

5.3.4 Cena para o Modelo

Imagens do mundo real com mundo virtual são capturadas para encontrar imagens no modelo e compará-las

5.4 Características Locais

Características locais são padrões em imagens que diferenciam de seu vizinho imediato, geralmente é caracterizada por mudanças em propriedades das imagens ou diversas propriedades simultaneamente. As propriedades mais comuns são intensidade, cor e texturas.

5.4.1 Propriedades da Característica Local ideal

Algoritmos de reconhecimento baseam-se em comparação de características recuperadas da cena. A recuperação e comparação de pontos tem um custo computacional relevante perto do tempo de execução da aplicação, portanto selecionar o menor número possível de características aumenta o desempenho e diminui o tempo de resposta da aplicação. Garantir que estamos selecionando boas características pode ser crucial na eficácia do reconhecimento. Segundo ([TINNE TUYTELAARS, 2007](#)) , boas características devem ter as seguintes propriedades:

- **Repetibilidade:** Dadas duas imagens do mesmo objeto ou cena, tomadas em con-

dições ou pontos de vista diferentes, uma porcentagem alta de características deve ser reconhecidas se estiverem visíveis.

- **Distinção:** Os padrões reconhecidos têm de ser possíveis de serem distinguidos entre si para facilitar o casamento.
- **Localidade:** As características devem ser locais para reduzir a probabilidade de oclusão.
- **Quantidade:** O número de pontos detectados tem que ser o suficiente para que mesmo objetos pequenos tenham minimamente características que possam ser localizados e para que o objeto possa sofrer oclusão e ainda assim seja reconhecido.
- **Acurácia:** As características detectadas tem que ser localizadas com o máximo de acurácia possível com respeito tanto referente à posição quanto à escala.
- **Eficiência:** De preferência a detecção deve ser o mais rápido possível.

5.4.2 Features

Antes de compreender como é feito o reconhecimento e registro de imagens é importante nos perguntar como nós conseguimos reconhecer objetos em uma cena, como conseguimos comparar facilmente objetos em duas imagens distintas. Somos treinados desde cedo a diferenciar formas geométricas, perceber escalas diferentes ou mesmo reconhecer o mesmo objeto independente de como está posicionado na cena buscando padrões que categorizem e diferenciem o objeto. Instintivamente conseguimos reconhecer boas características e localizar objetos. Na imagem 5.4 temos uma imagem de um prédio e seis recortes, dos quais conseguimos facilmente reconhecer com precisão a de letra E e F, as

de letra A,B,C,D podemos identificar possíveis localizações mas não podemos dizer com certeza onde estão na imagem.



FIGURA 5.4 – Reconhecimento de padrões.

As características E e F são o que chamamos de good features pois o nível de certeza é bem alto.

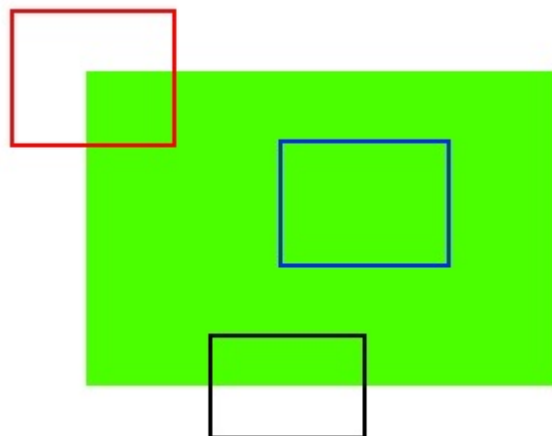


FIGURA 5.5 – Regiões de reconhecimento de padrões.

A Imagem 5.5 ilustra os tipos de características. A região azul não possibilita diferenciar onde está na imagem, a região preta pode ser confundida com qualquer uma das regiões ao deslocarmos horizontalmente, a imagem vermelha nos possibilita diferenciar e reconhecer o canto da imagem verde com precisão milimétrica. Podemos então concluir que uma característica é boa para ser utilizada como parâmetro de entrada para algoritmos de reconhecimento, quanto maior foi o nível de certeza da sua localização, o que facilita o **Feature Detection**. Para localizar o mesmo objeto em outra imagem é necessário identificarmos a região onde se encontra, caso contrário no exemplo da imagem 5.4 seria impossível localizar uma janela específica. Tal descrição de contexto é chamada de **Feature Description**. Uma vez de posse da característica e do seu contexto é possível reconhecer o objeto de fato.

5.5 Cadência

É a medida do número de quadros individuais que um determinado dispositivo óptico ou eletrônico processa e exibe por unidade de tempo. Em geral a cadência é medida em fps. Em cinema a cadência de projeção padrão desde 1929 foi fixada em 24fps mas no período do cinema mudo a maioria dos filmes eram rodados com cadência entre 16 e 20fps. Em vídeo, os principais sistemas lidam com cadência entre 25fps(PAL) e 30fps(NTSC). As aplicações devem ter cadência tolerável dependendo de seu uso, segundo (TANG, 1993) para aplicações interativas o mínimo tolerável é de 5fps enquanto para aplicações de animações fluidas de 30fps. Sendo a cadência, a frequência entre frames, deve ser contabilizado o tempo de gerar a informação e o tempo de dispor a informação no dispositivo óptico. O tempo de cada frame é calculado por

$$t_{frame} = \frac{1}{fps}$$

No caso de cadência mínima de 5fps, temos quadros com tempo menor que 200ms, portanto as análises devem ser balisadas a tempos menores.

5.6 Camera

Como descrito em ([GARY BRADSKI, 2008](#)) as câmeras de baixo custo vendidas atualmente introduzem diversos erros de distorção. Dois dos maiores são as distorções radiais e as distorções tangenciais. Devido a distorções radiais, linhas retas vão parecer curvadas e tal efeito tende a aumentar quanto mais o ponto se distancia do centro da imagem. Na prática essas distorções são pequenas e podem ser caracterizadas pelos primeiros termos de uma expansão da série de Taylor.

5.6.1 Distorção Radial

Distorções radiais são mais associadas a lentes de grandes ângulos, em particular zoom wide-angles. Esse efeito resulta em imagens esforçadas, o que significa que as arestas da imagem vão aparecer curvadas a olho humano. A distorção é resultado da fabricação por ser bem mais barato e prático produzir lentes esféricas do que lentes mais matematicamente ideais com perfis parabólicos.

A imagem [5.7](#) mostra como o deslocamento de um retângulo devido à distorção radial, mais acentuado quanto mais próximo das bordas da imagem.

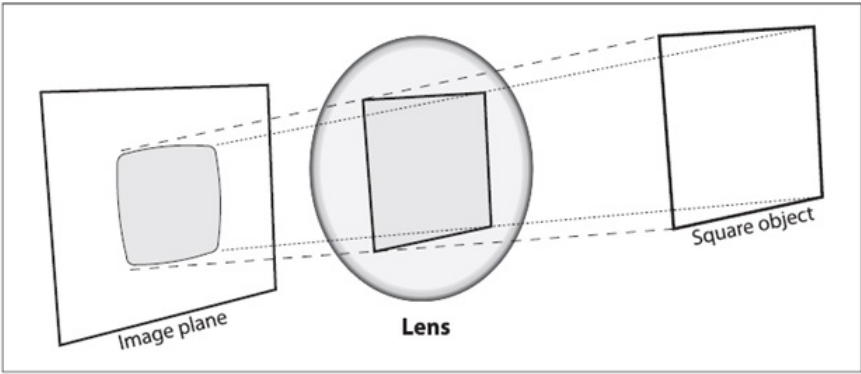


FIGURA 5.6 – Distorção radial.

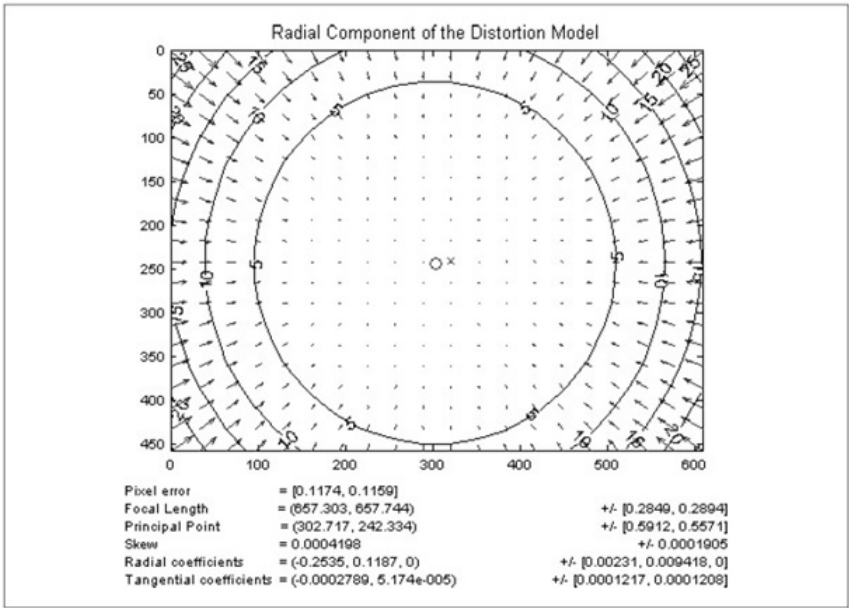


FIGURA 5.7 – Componentes de Distorção radial.

5.6.2 Distorção Tangencial

Essa distorção se deve a defeitos de manufatura causadas pelo fato de o plano da lente não ser totalmente paralelo ao plano da imagem.

A imagem 5.9 mostra o efeito da distorção tangencial em um grid de pontos retangular. Os pontos se deslocam de forma elíptica quanto mais próximos da borda.

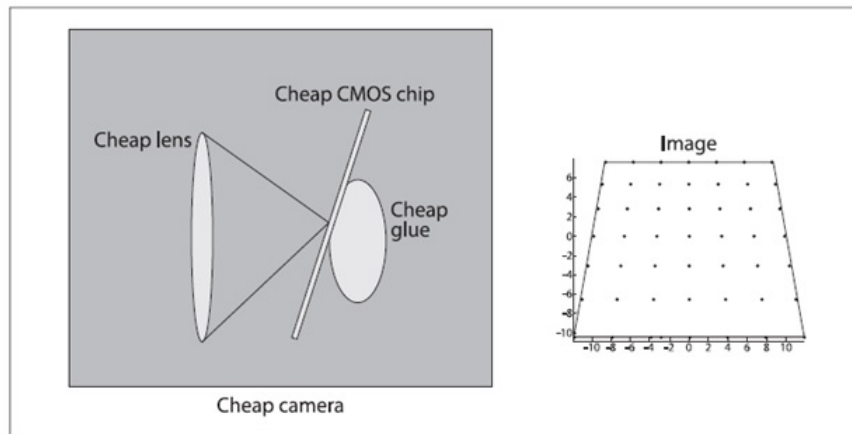


FIGURA 5.8 – Distorção tangencial.

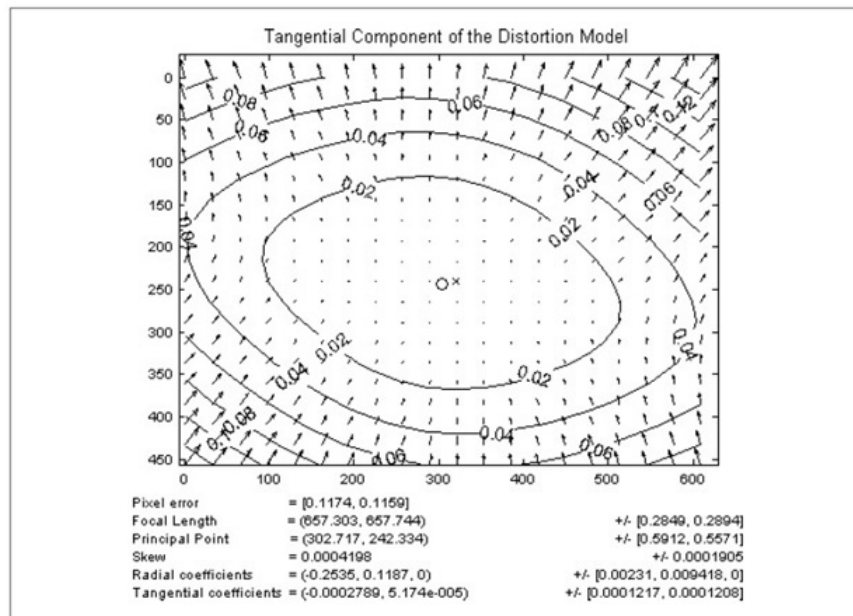


FIGURA 5.9 – Componentes de Distorção tangencial.

5.7 Modelo de Câmera

As câmeras são modeladas como câmeras de pequenos orifícios como mostrada na imagem 5.10. Esse modelo define a projeção básica sobre a qual as imagens 2D serão mapeadas. Esse modelo é uma simplificação óptica de uma câmera real e é utilizado comumente para descrever a formação de imagens na câmera. O sistema de coordenadas considerado é a convenção da mão direita com centro de coordenadas de projeção na origem e a imagem a uma distância focal f . ref[3]

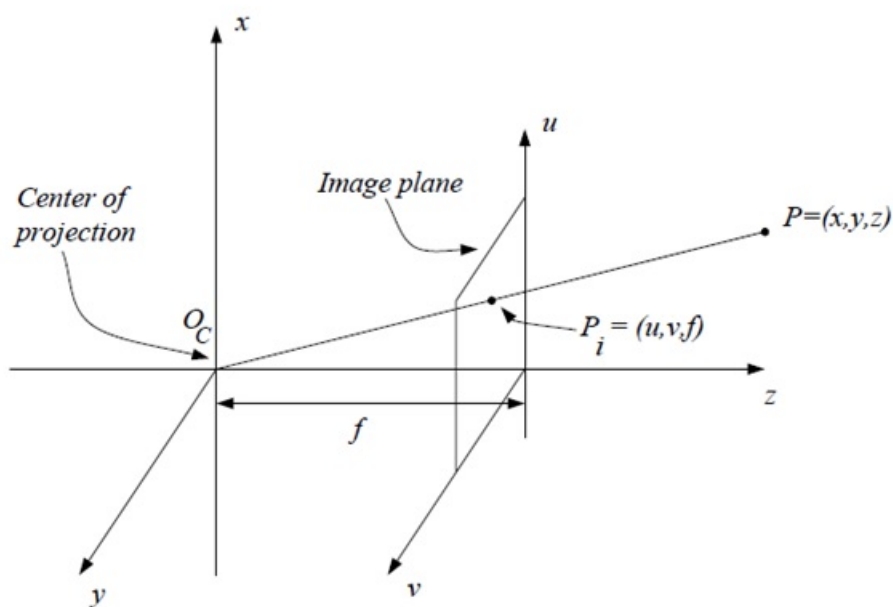


FIGURA 5.10 – Camera no espaço vetorial.

5.8 Reconhecimento

É a etapa em que padrões são identificados e comparados para posteriormente identificar objetos

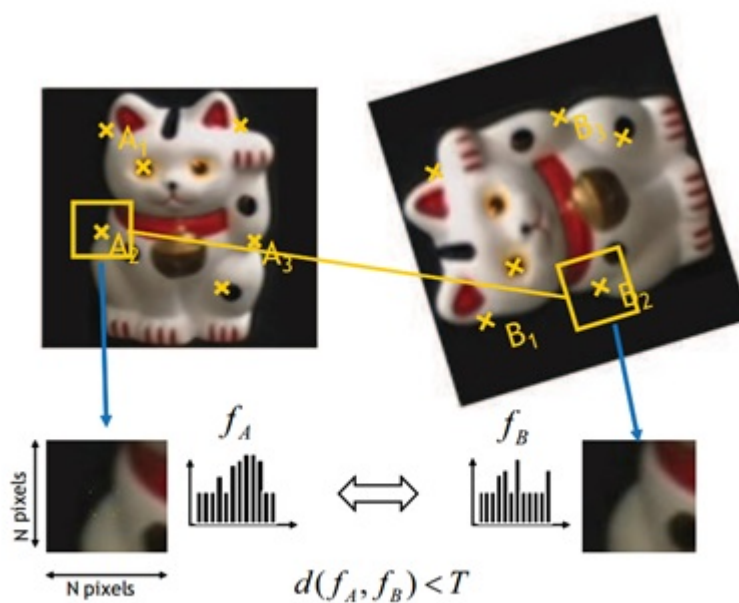


FIGURA 5.11 – Ilustração do procedimento de reconhecimento com features locais.

Pipeline de reconhecimento como ilustrado na imagem [5.11](#) :

- Encontrar um grupo de keypoints distintos
- Definir uma região em torno de cada keypoint
- Extrair e normalizar o conteúdo da região
- Calcular um descritor para a região normalizada
- Encontrar correspondências de descritores.

5.8.1 Detecção de características

O primeiro passo para o reconhecimento de objetos é a detecção de características. Para que possa ser feita a comparação da imagem referencia com a imagem sentida. A abstração de informações a ser reconhecida tem que ser suficiente para lidar com escalas diferentes, rotações entre as imagens, e pequenas distorções. Para que o reconhecimento seja eficiente e eficaz são necessários que uma massa de pontos mínima seja selecionada e feita a correspondência entre as imagens sentidas e referência. As características utilizadas são em geral cantos, linhas, curvas, padrões ou regiões. O tipo de característica selecionada é dependente do tipo de imagem provida. Imagens de cenas feitas a mão geralmente são compostas de segmentos de retas enquanto imagens de satélite são geralmente compostas de contornos e regiões. Quanto mais invariantes forem as características encontradas, mais robusto e preciso é o processo de comparação.

5.8.2 Correspondência de características

A fase de correspondência de características é feita tanto selecionando características na imagem referência e procurando a correspondente na imagem sentida ou mesmo selecionando características nas duas imagens independentemente e procurando a correspondência entre elas. Quando a característica selecionada não for do tipo ponto, é importante para cada par de correspondências, pelo menos um ponto ser determinado para que seja utilizado para determinar posteriormente os parâmetros de transformação. Por exemplo, se forem selecionados padrões como tipo de característica, o centro do padrão é considerado o ponto, se for selecionado uma região, o centro de massa da região é o ponto de apoio, se linhas forem tomadas como tipo de característica, devem ser tomados intersecções como ponto de apoio e finalmente se forem selecionadas curvas, os máximos locais são considerados os pontos correspondentes.

5.9 Algoritmos de Reconhecimento

Existem diversos algoritmos de reconhecimento de características, entretanto nesse artigo os testes serão restritos aos principais algoritmos Fast, GoodFeaturesToTrack, Mser, Star, Sift, Surf

5.9.1 BRISK

([LEUTENEGGER; CHLI; SIEGWART, 2011](#))

5.9.2 FAST

Segundo (ROSTEN; DRUMMOND, 2006), é um método de reconhecimento baseado em detecção de arestas originalmente desenvolvido por Edward Rosten e Tom Drummond no seu paper Machine learning for high-speed corner detection in 2006 (Later revised it in 2010) A maior promessa do método é a eficiência computacional O método considera um círculo de dezesseis pixels ao redor da aresta considerada p . O detector original (ROSTEN; DRUMMOND, 2005),(E. ROSTEN; DRUMMOND, 2005) classifica p como uma aresta se existem n pixels contíguos em um círculo que são mais brilhantes do que o pixel candidato de intensidade I_p mais um threshold t ou mais escuros do que $I_p - t$

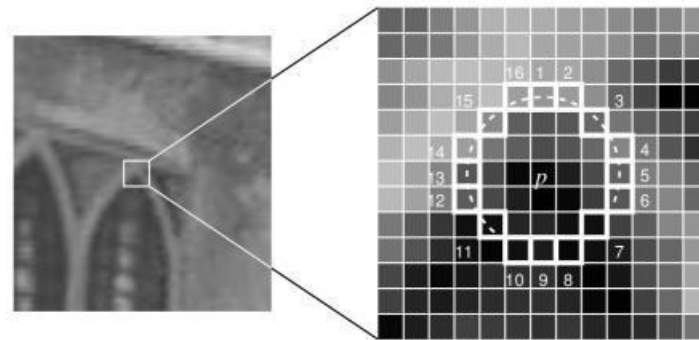


FIGURA 5.12 – FAST

Na imagem 5.12 foi escolhido $n=12$. Um teste rápido é

1. selecionar o ponto é testar primeiro as extremidades. no caso da imagem, escolhido o ponto p
2. Compara-se os pontos 1 e 9, e verifica-se se o ponto p tem intensidade com diferença de módulo t , ou seja os pontos 1 e 9 são mais claros ou mais escuros do que o ponto p pelo fator de t
3. Caso o ponto p continue sendo um candidato, avalia-se os pontos 5 e 13

4. Se p é uma aresta então pelo menos 3 desses pontos devem ser mais brilhantes ou mais escuros do que p então o teste pode ser feito nos demais pontos.

5.9.3 FREAK

([ALEXANDRE ALAHI RAPHAEL ORTIZ, 2012](#))

5.9.4 MSER

5.9.5 ORB

([RUBLEE, 2011](#))

5.9.6 SIFT - Scale-Invariant Feature Transform

De acordo com ([LOWE, 2004](#)) é um método baseado em detecção de arestas que tem como proposta garantir a invariância a variação de escala. Um problema que nos métodos de reconhecimento de arestas se não tratado pode causar diminuição na robustez do algoritmo. A figura 5.13 ilustra bem o efeito que a mudança de escala pode fazer.

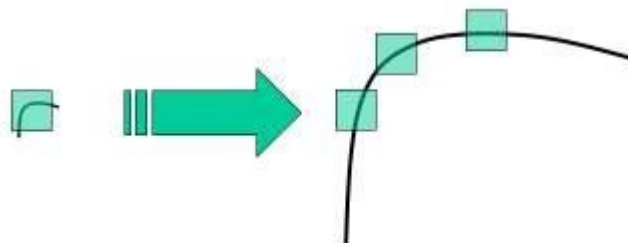


FIGURA 5.13 – SIFT

Em 2004, D.Lowe, descreveu o algoritmo Scale Invariant Feature Transform no seu paper Distinctive Image Features from Scale-Invariant Keypoints[15] como extrair key-

points e computar os descritores. 8.10.2.1. Scale-space Extrema Detection Observando a imagem 5.13 é possível notar que não podemos utilizar a mesma janela de inspeção independente da escala do objeto, para objetos maiores temos que utilizar janelas maiores. Nesse contexto, o filtro scale-space é utilizado e são calculados Laplacianos de Gaucianos com diversos valores de σ . Os LoG funcionam como blob detectors que detectam blobs em várias escalas com a variação de σ .

Utilizar Laplacianos de Gaucianos é uma abordagem custosa computacionalmente, como uma forma de aproximação são utilizados diferenças de Gaucianas.

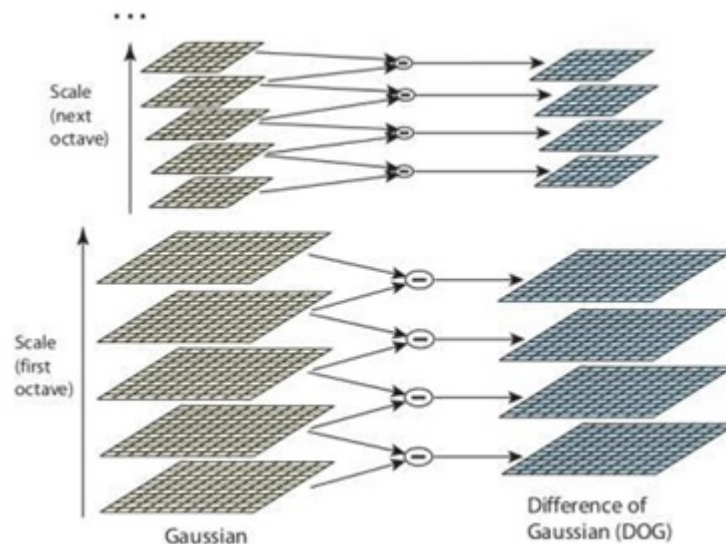


FIGURA 5.14 – Difference of Gaussian

Uma vez que as diferenças de gaussianas são calculadas, é necessário procurar por máximos entre espaço e escalas diferentes. Por exemplo como mostrado na figure 5.15, um pixel na imagem é comparado com seus 8 vizinhos e com níveis de escala próximos e anteriores. O que representa que o keypoint encontrado é melhor representado naquela escala.

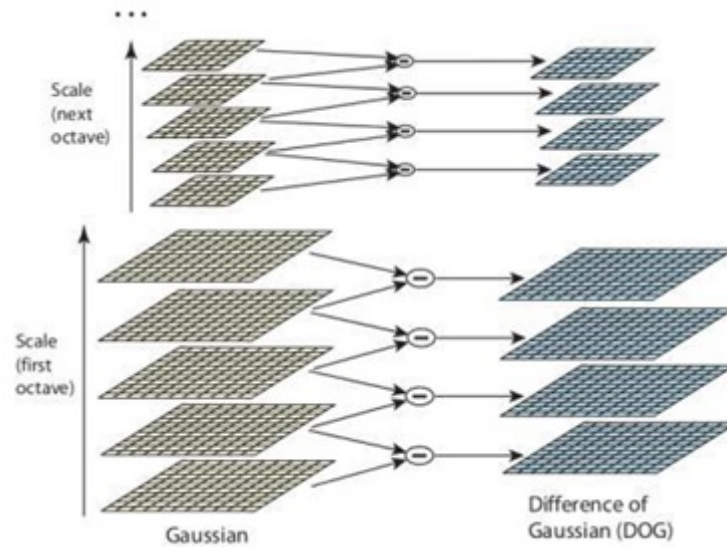


FIGURA 5.15 – Difference of Gaussian

5.9.6.1 Keypoint Localization

Uma vez que keypoints potenciais são localizados é importante selecionar pontos de interesse com contraste alto. A localização dos pontos é refinada utilizando uma expansão de Taylor e se as intensidades dos máximos forem menores do que um threshold são rejeitados. Uma característica do DoG é a alta resposta a arestas gerando falsos positivos. Portanto é necessário eliminar algumas arestas identificadas erroneamente.

5.9.6.2 Orientation Assignment

TODO

5.9.6.3 Keypoint Descriptor

TODO

5.9.6.4 Keypoint Matching

O casamento entre dois pontos é feito identificando pontos próximos. Entretanto em algumas situações existem pontos muito próximos que podem ser causados por ruídos na detecção de pontos de interesse, nesse caso é calculada uma razão de distância entre o ponto de interesse com o mais próximo, e com o segundo mais próximo, se a razão for maior do que 80 são rejeitados. Tal abordagem elimina cerca de 90 de falsos positivos e 5 de pontos corretos.

5.9.7 STAR

5.9.8 SURF - Speeded Up Robust Feature

Segundo ([HERBERT BAY TINNE TUYTELAARS, 2008](#)) em comparação com o algoritmo SIFT que aproxima Laplacianas de Gaussianas por Diferenças de Gaussianas (DoG), SURF aproxima LoG de Box-type Filter e não é utilizado nenhum tipo de suavização entre escalas, o que garante mais agilidade nos resultados porque a convolução com box filters são muito mais rápidos com o uso de integral images e pode ser paralelizado

Em geral SURF se apresenta cinco vezes mais rápido do que DoG.

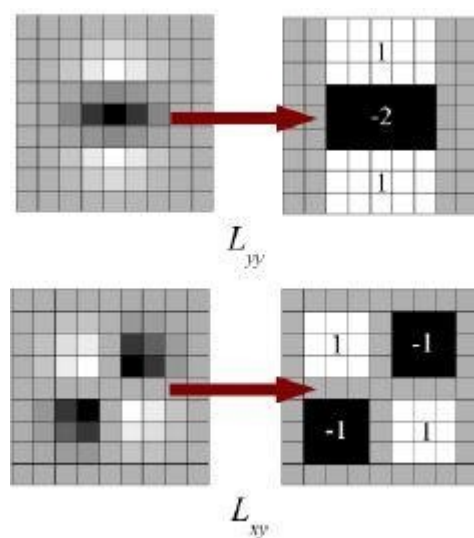


FIGURA 5.16 – Box Filtering

6 Metodologia

6.1 Definição de parâmetros

Os testes serão realizados adicionando variáveis de forma artificial por meio de transformações afins de forma a emular algum comportamento ou situação encontrada no ambiente de manutenção

- Variação de escala para simular a aproximação dentro da janela de inspeção
- Variação de rotação para simular a movimentação durante a manutenção
- Variação de iluminação para simular manutenções feitas em horários do dia e iluminação diferente como por exemplo ambientes com neve com brilho muito maior
- Adição de Blur para emular ambientes com muita poeira ou esfumaçados

A definição das faixas de parâmetro utilizados foi feita com inspeção em campo

6.2 Análise de dados

A análise de desempenho de cada técnica deve ser balizada por parâmetros que garantam que a aplicação seja viável para o contexto citado na seção [3](#). Para tal deve-se

considerar os parâmetros estimados abaixo definidos por inspeção

- Cadência $> 5\text{fps}$ ou seja, tempo $< 200\text{ms}$ ([TANG, 1993](#))
- Robustez à escala: $0.5 < \text{escala} < 1.2$
- Robustez à rotação: $0 < \text{rotação} < 30$
- Robustez à iluminação: $-50 < \text{iluminação} < 50$
- Robustez à blur: $\text{kernelsize} < 4$

Uma análise mais precisa dos parâmetros não faz parte do escopo desse trabalho pois envolve pesquisa de campo em situações adversas. Portanto tais parâmetros foram escolhidos para validar o método de seleção das técnicas.

7 Protótipos

O escopo de desenvolvimento dessa tese tem por traçar um método de escolha do algoritmo a ser utilizado dependente de aspectos descritos na seção 3.1, os testes serão feitos em duas etapas: Definição de parâmetros de contorno e avaliação dos algoritmos mais conhecidos no contexto aeronáutico

7.1 Testes de reconhecimento de padrões

Utilizando os parâmetros definidos na etapa anterior serão rodados todos os algoritmos para avaliar qual se encaixa melhor em cada uma das situações. O protótipo para testes será realizado utilizando o OpenCV por já disponibilizar os algoritmos mais utilizados para o reconhecimento de padrões e algoritmos de visão computacional. Nesse protótipo será avaliado a aderência dos algoritmos: Fast, GoodFeaturesToTrack, Mser, Star, Sift, Surf, Brisk. Segundo os seguintes critérios:

Velocidade - para garantir que a aplicação poderá ser utilizada como uma ferramenta de tempo real é imprescindível buscar uma aplicação que rode naturalmente a 30 frames por segundo. Tal realidade é facilmente atingida em computadores com vários núcleos como os core i7 mas é importante lembrar que o cenário de realidade aumentada para ter um contexto coerente com a proposta de imersão é composto por dispositivos portáteis

que via de realidade regra não tem poder de processamento tão alto.

Qualidade - características detectadas são geralmente utilizadas posteriormente em etapas de rastrio e casamento de padrão. Para os testes de rastreio serão adicionadas às imagens algumas transformações afins, como escala, rotação e iluminação e então estimar a qualidade das características.

Iluminação e invariância à escala - Detectores de características tem que ter a habilidade de reconhecer características independente do tamanho do objeto. A invariância deve ser verdadeira para variação de iluminação. Variações pequenas de iluminação e contraste não devem afetar o detector significativamente. As câmeras atuais em geral tem controle automático de ganho que ajusta automaticamente a exposição evitando sub ou super exposição.

Serão utilizados as seguintes transformações que tem por finalidade avaliar as variações de ambiente

- **Rotação:** Robustez à rotação também é uma característica importante pois nem sempre encontraremos situações em que o dispositivo de RA será posicionado na mesma orientação o tempo todo, garantindo uma maior autonomia a quem executa a manutenção
- **Brilho:** A variação de brilho pode emular tanto variações temporais, como dia e noite, mas também pode emular situações de muita luminosidade como neve por exemplo
- **Blur:** A robustez a Blur auxilia na resposta do algoritmo frente à movimentos bruscos
- **Escala:** A robustez à escala do objeto garante que o mesmo será reconhecido tanto

ao longe quanto em situações bem próximas

7.2 Ambiente

Os testes tem por objetivo garantir as restrições descritas na seção 3. Constraints, portanto é o desempenho e o tempo de reconhecimento são uma característica relevante. Os resultados obtidos pelos testes descritos nessa tese foram executado em uma máquina com a seguinte configuração:

Processador: Core 2-duo 2.2GHz

Memória: 4GB

Placa de Vídeo: NVidia GForce 9300M GS

Biblioteca: OpenCV 2.4.10

7.2.1 OpenCV

Distribuído sob licença BSD e portanto livre para uso acadêmico e comercial. Possui interfaces para C++, C, Python e Java suportando Windows, Linux, Mac OS, iOS e Android. OpenCV foi desenvolvido para resultados eficientes e com foco em aplicações de tempo real, podendo tomar vantagem de processamentos paralelos, utilizando-se de OpenCL aproveitando-se da aceleração de hardware ou mesmo de plataformas heterogêneas. Adotado ao redor do mundo em várias pesquisas com uma comunidade de mais de 47 mil pessoas com mais de 9 milhões de downloads. Adotado como plataforma de desenvolvimento para aplicações de arte interativa a aplicações de robótica avançada.

8 Resultados

A abordagem de testes e análise de desempenho é feita como descrito em ([KRYS-TIAN MIKOLAJCZYK, 2005](#)).

Para que sejam respeitados as restrições citadas na seção [3](#) e para que a aplicação seja coerente para uso cotidiano no contexto de manutenção de aeronaves, a velocidade de reconhecimento é muito importante.

8.1 Análise de parâmetros pontuais

As implementações existentes no OpenCV para as técnicas utilizadas nessa tese tem parâmetros configuráveis, o que pode garantir uma maior aderência às necessidades dependendo do contexto, portanto antes de compararmos as técnicas entre si é importante garantirmos que a configuração utilizada de cada uma se encaixa nas restrições citadas em [3](#). Cada técnica possibilita variar alguns parâmetros que será analisado para encontrar a configuração mais adequada.

8.1.1 BRISK

A assinatura do método é

BRISK(int thresh=30, int octaves=3, float patternScale=1.0f);

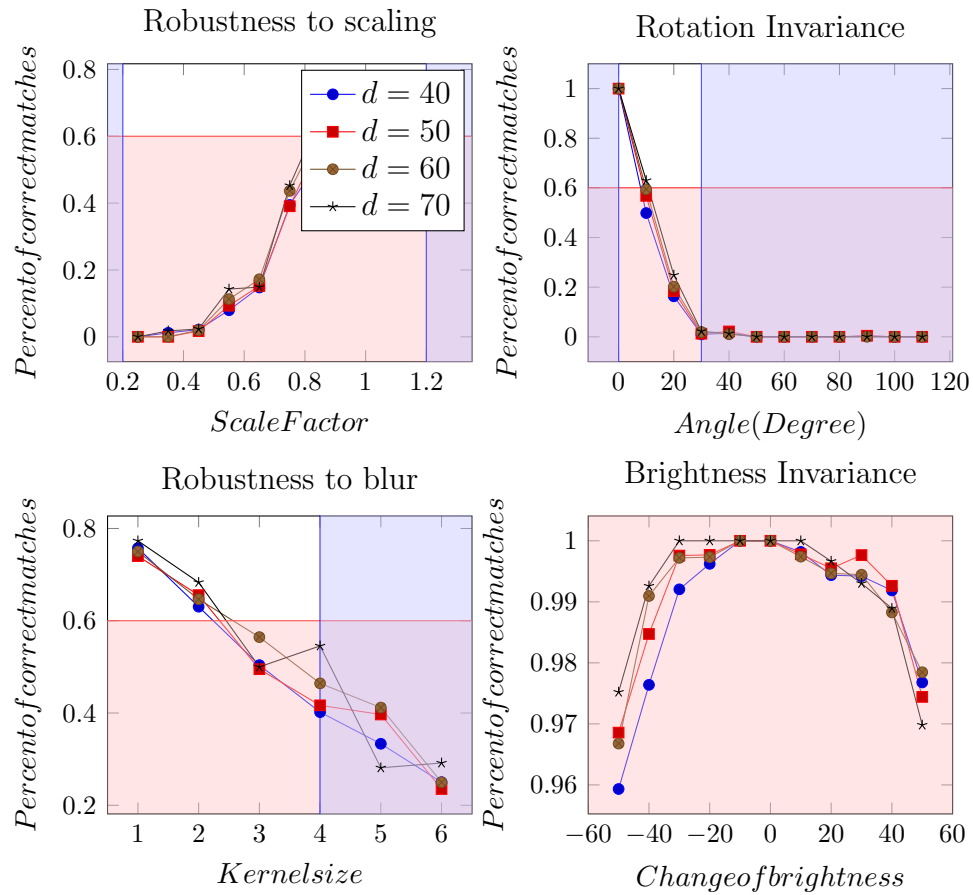


FIGURA 8.1 – Resultado de testes individuais de performance com BRISK

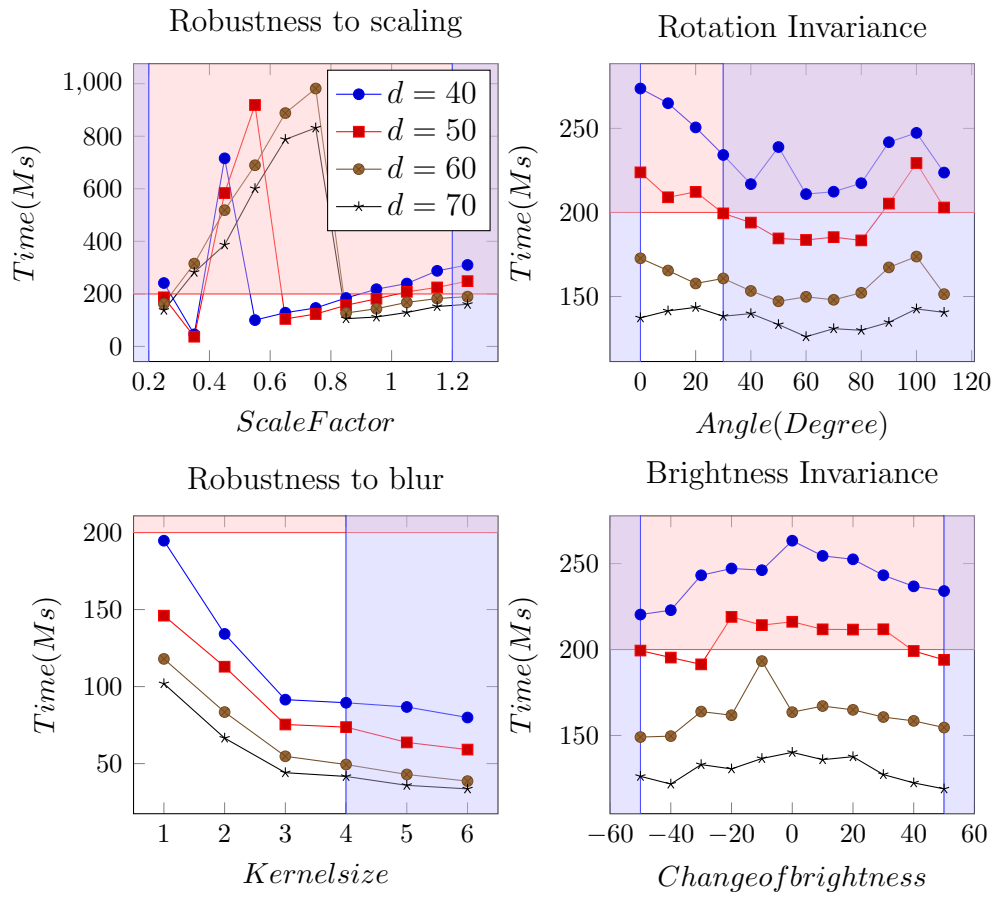


FIGURA 8.2 – Resultado de testes individuais de tempo com BRISK

8.1.1.1 Análise

Fazendo uma análise de tendência variando o parâmetro thresh de 10 à 40 podemos inferir que A taxa de acertos:

- escala é diretamente proporcional
- rotação é inversamente proporcional
- blur é diretamente proporcional
- iluminação é diretamente proporcional

O tempo de execução todos os parâmetros são inversamente proporcionais

8.1.2 FAST

A assinatura do método é

```
FastFeatureDetector( int threshold=1, bool nonmaxSuppression=true );
```

variando o parâmetro treshhold em 10,20,30,40

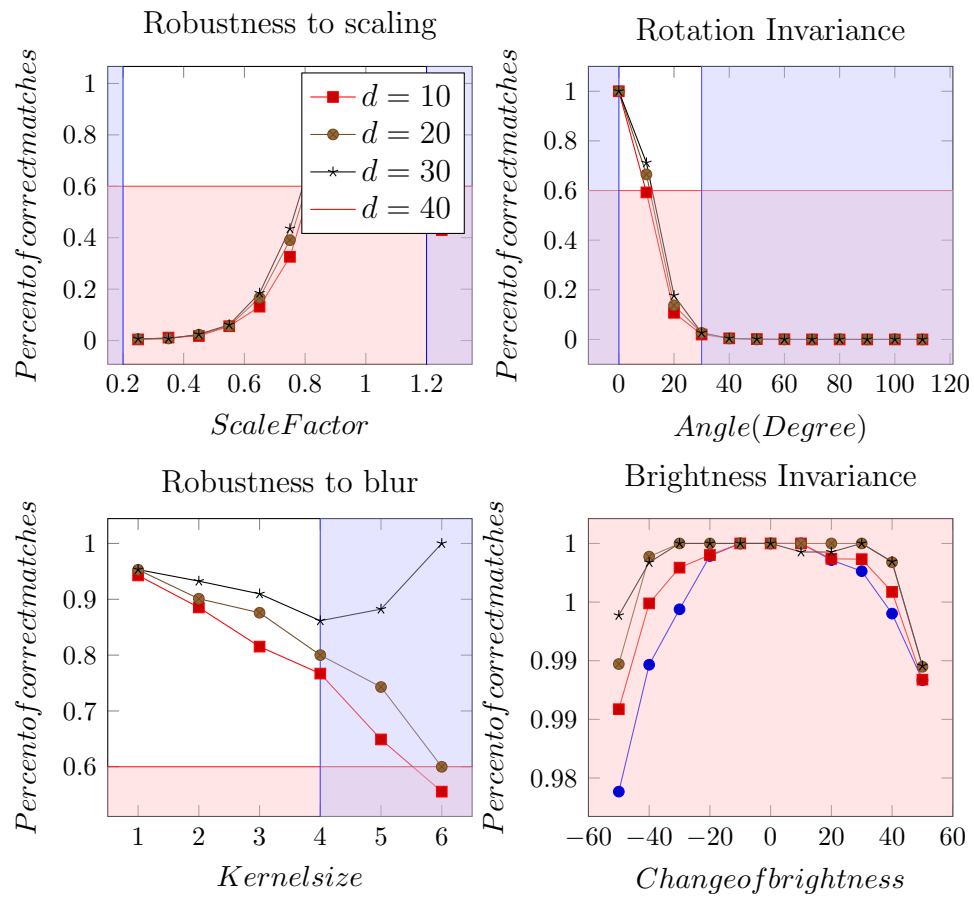


FIGURA 8.3 – Resultado de testes individuais de performance com FAST

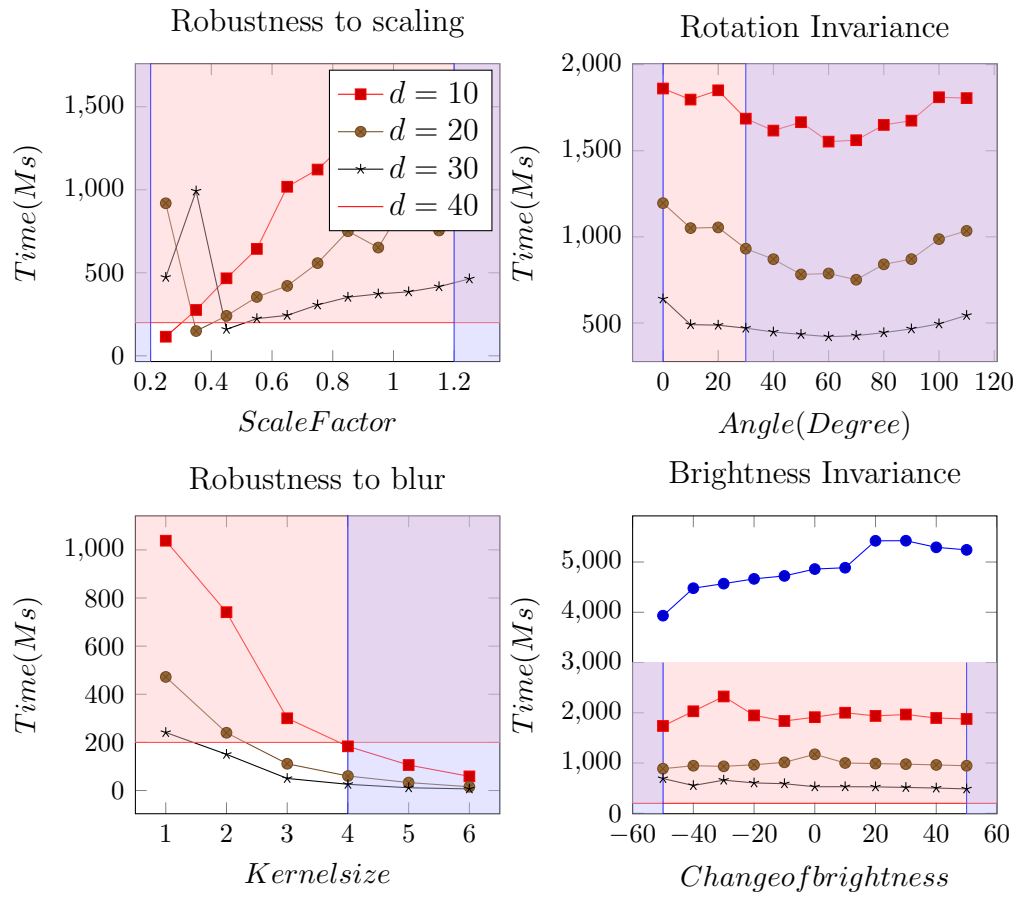


FIGURA 8.4 – Resultado de testes individuais de tempo com FAST

8.1.3 FREAK

A assinatura do método é

TODO

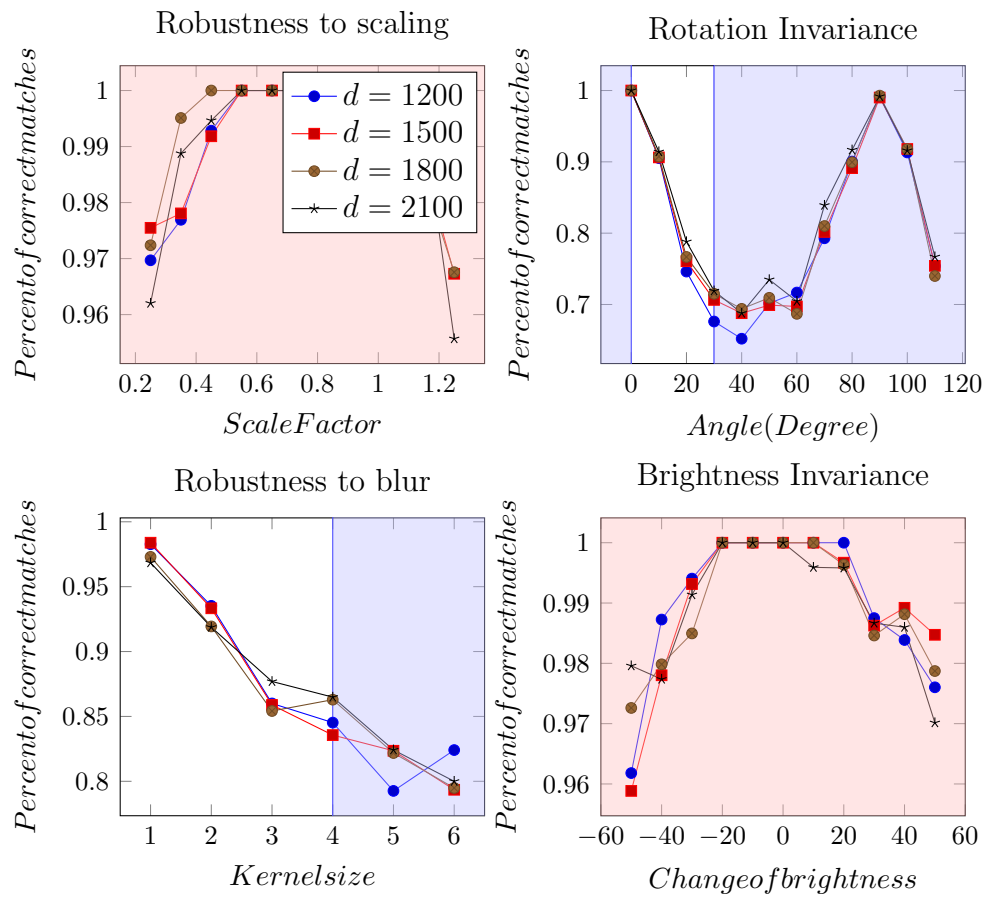


FIGURA 8.5 – Resultado de testes individuais de performance com FREAK

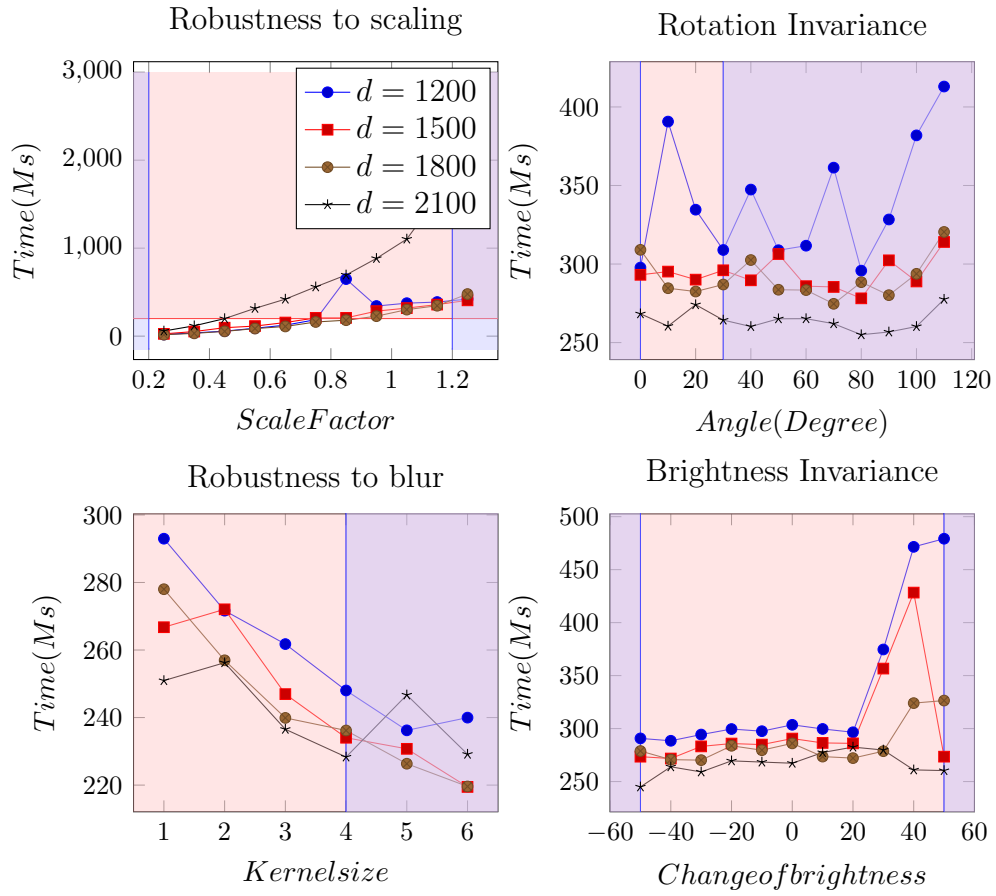


FIGURA 8.6 – Resultado de testes individuais de tempo com FREAK

8.1.4 MSER

A assinatura do método é

```
MSER( int  $\delta$  = 5, int  $m_{in\_area}$  = 60, int  $m_{ax\_area}$  = 14400, double  $m_{ax\_variation}$  =
0.25, double  $m_{in\_diversity}$  = .2, int  $m_{ax\_evolution}$  = 200, double  $a_{req\_threshold}$  = 1.01, double  $m_{in\_margin}$ 
0.003, int  $d_{ge\_blur\_size}$  = 5);
```

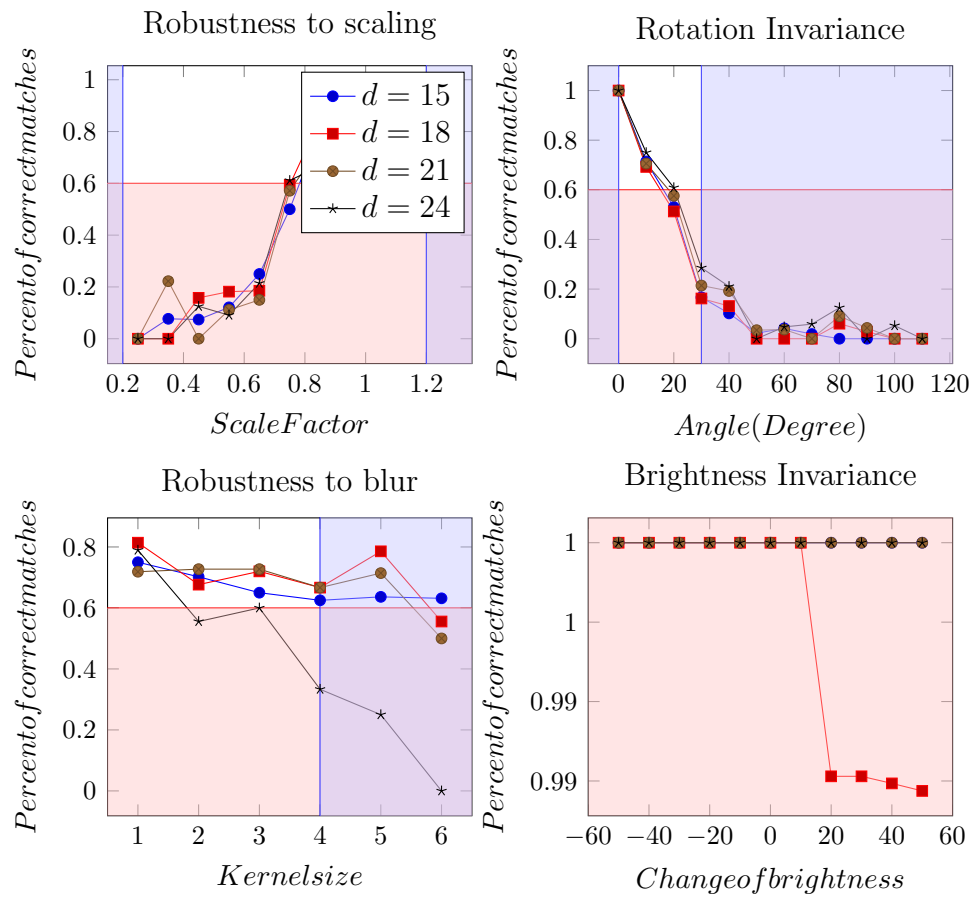


FIGURA 8.7 – Resultado de testes individuais de performance com MSER

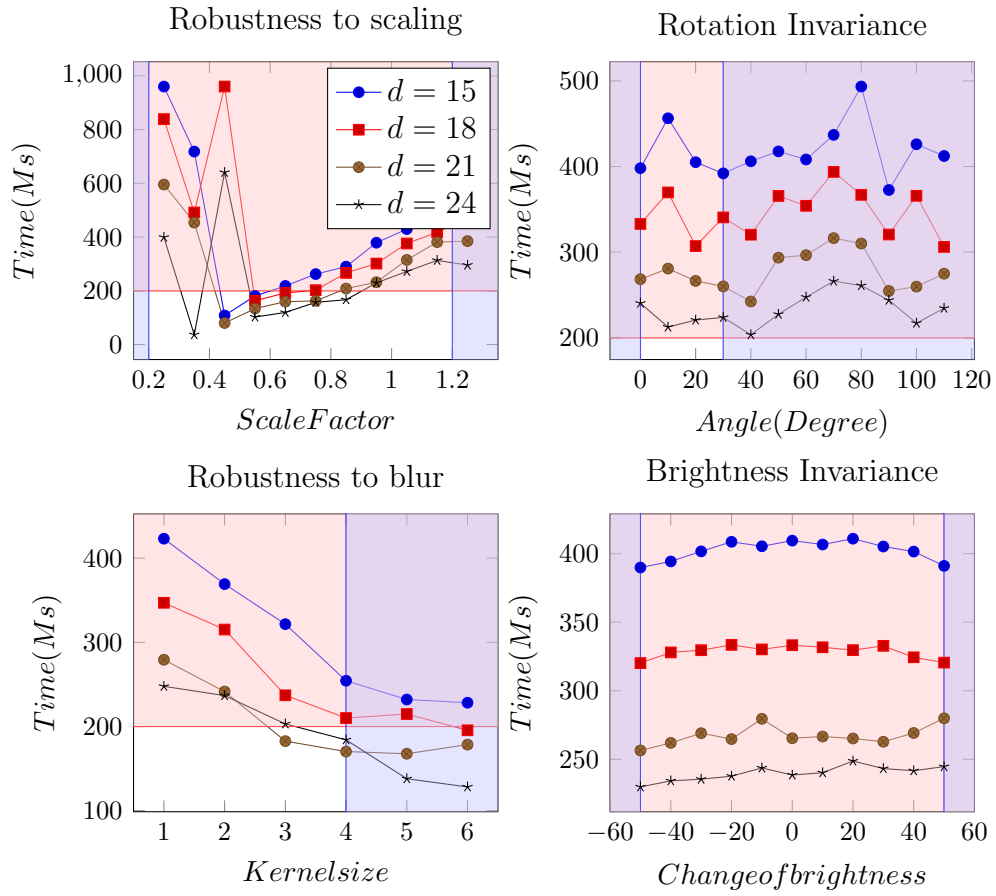


FIGURA 8.8 – Resultado de testes individuais de tempo com MSER

8.1.5 ORB

A assinatura do método é

```
ORB(int nfeatures = 500, float scaleFactor = 1.2f, int nlevels = 8, int edgeThreshold = 31,
int firstLevel = 0, int WTAK = 2, int scoreType = ORB :: HARRISSCORE, int patchSize =
31);
```

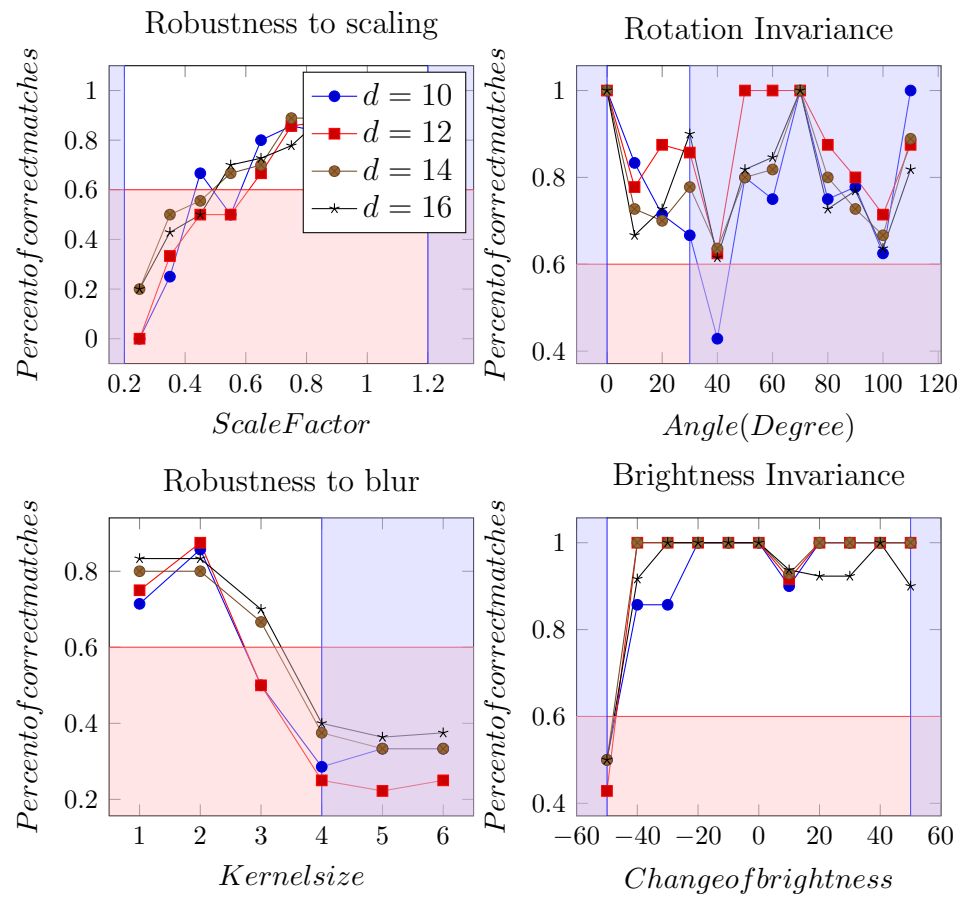



FIGURA 8.9 – Resultado de testes individuais de performance com ORB

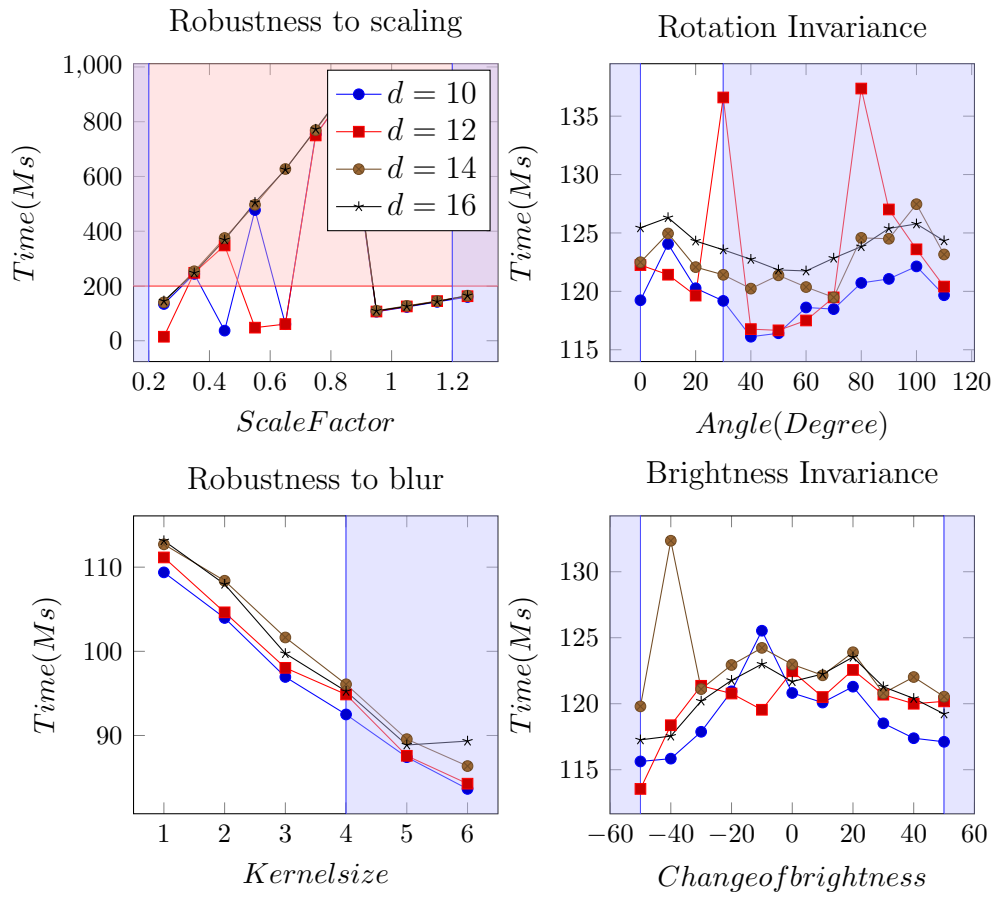


FIGURA 8.10 – Resultado de testes individuais de tempo com ORB

8.1.6 SIFT

A assinatura do método é

```
SIFT( int nfeatures=0, int nOctaveLayers=3, double contrastThreshold=0.04, double
      edgeThreshold=10, double sigma=1.6);
```

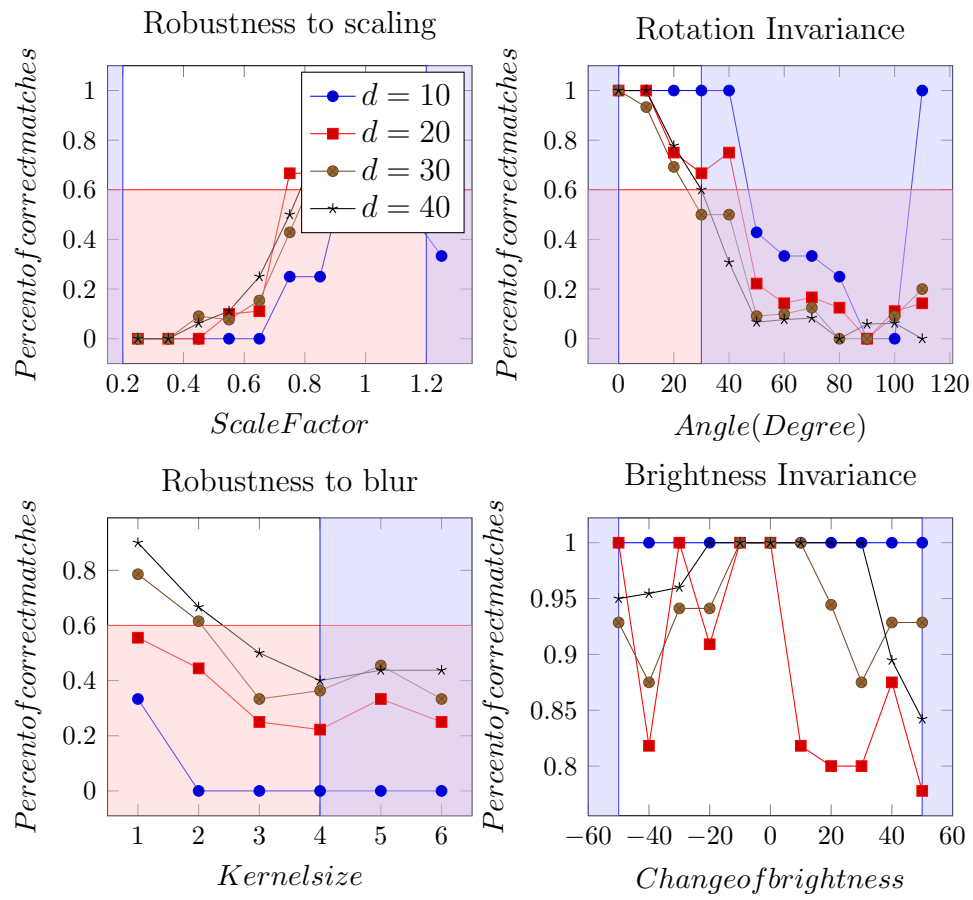


FIGURA 8.11 – Resultado de testes individuais de performance com SIFT

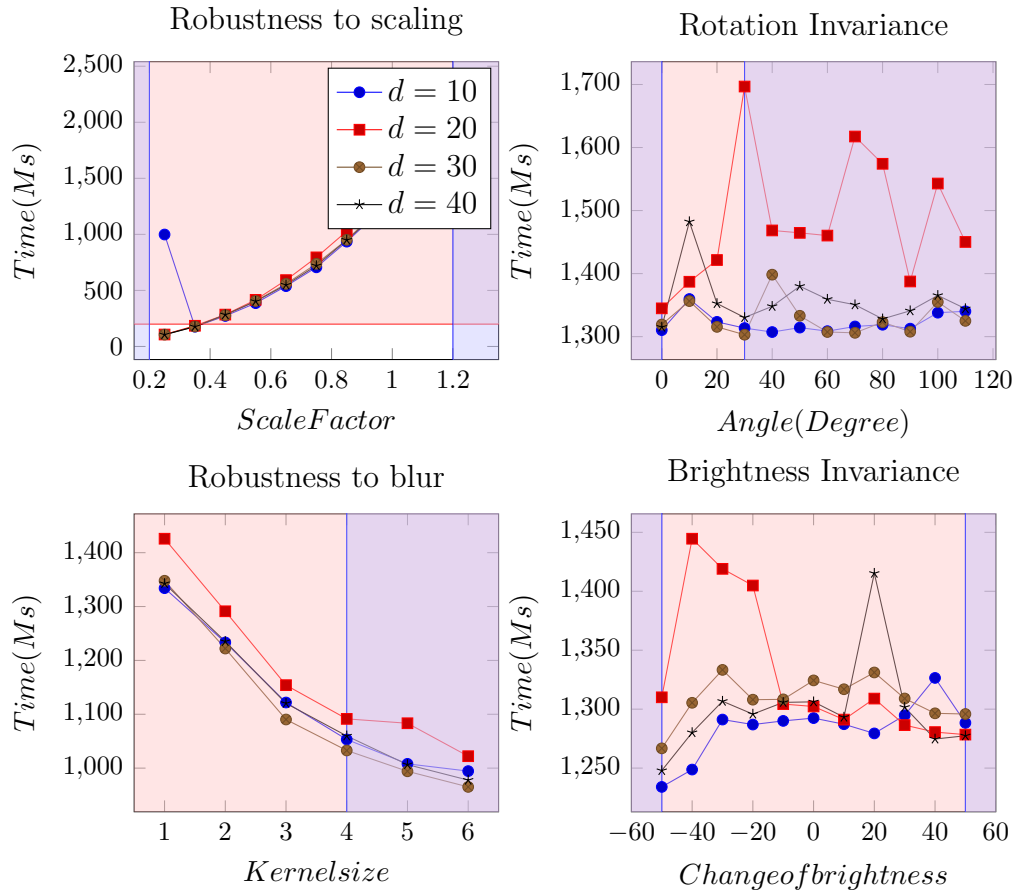


FIGURA 8.12 – Resultado de testes individuais de tempo com SIFT

8.1.7 STAR

A assinatura do método é

```
StarDetector(intmaxSize = 45, intresponseThreshold = 30, intlineThresholdProjected =
10, intlineThresholdBinarized = 8, intsuppressNonmaxSize = 5);
```

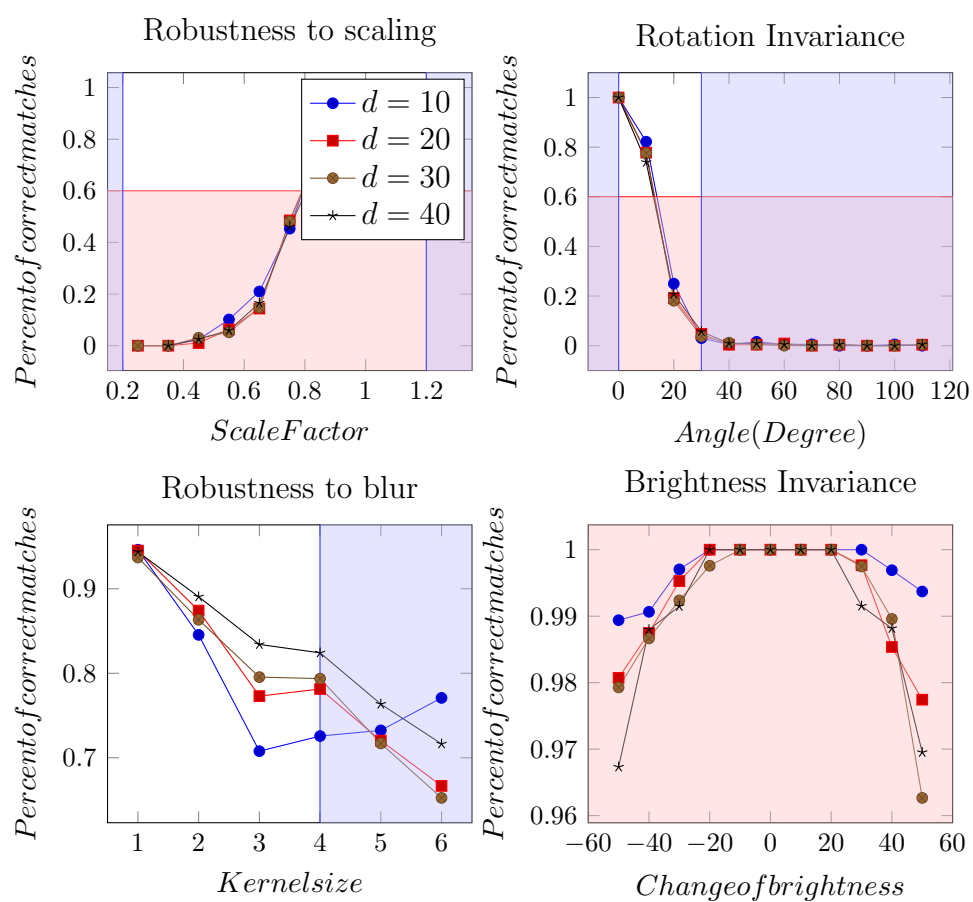


FIGURA 8.13 – Resultado de testes individuais de performance com STAR

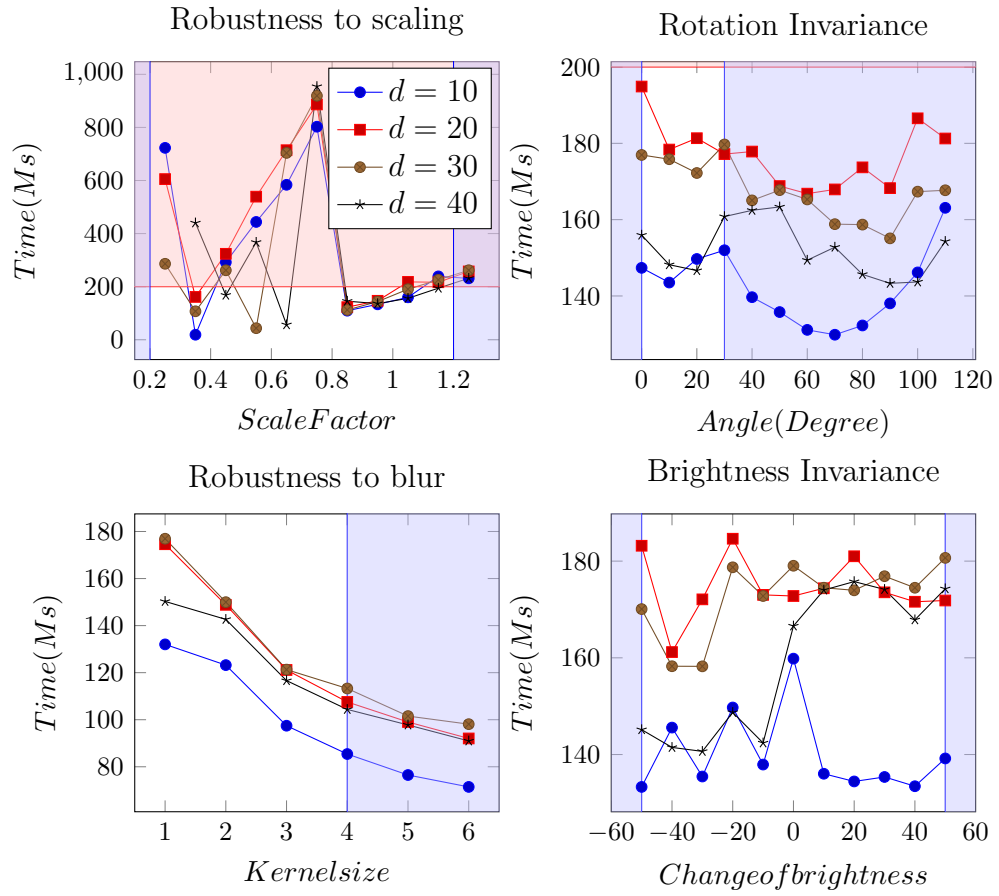


FIGURA 8.14 – Resultado de testes individuais de tempo com STAR

8.1.8 SURF

A assinatura do método é

```
SURFDetector(intmaxSize = 45, intresponseThreshold = 30, intlineThresholdProjected = 10, intlineThresholdBinarized = 8, intsuppressNonmaxSize = 5);
```

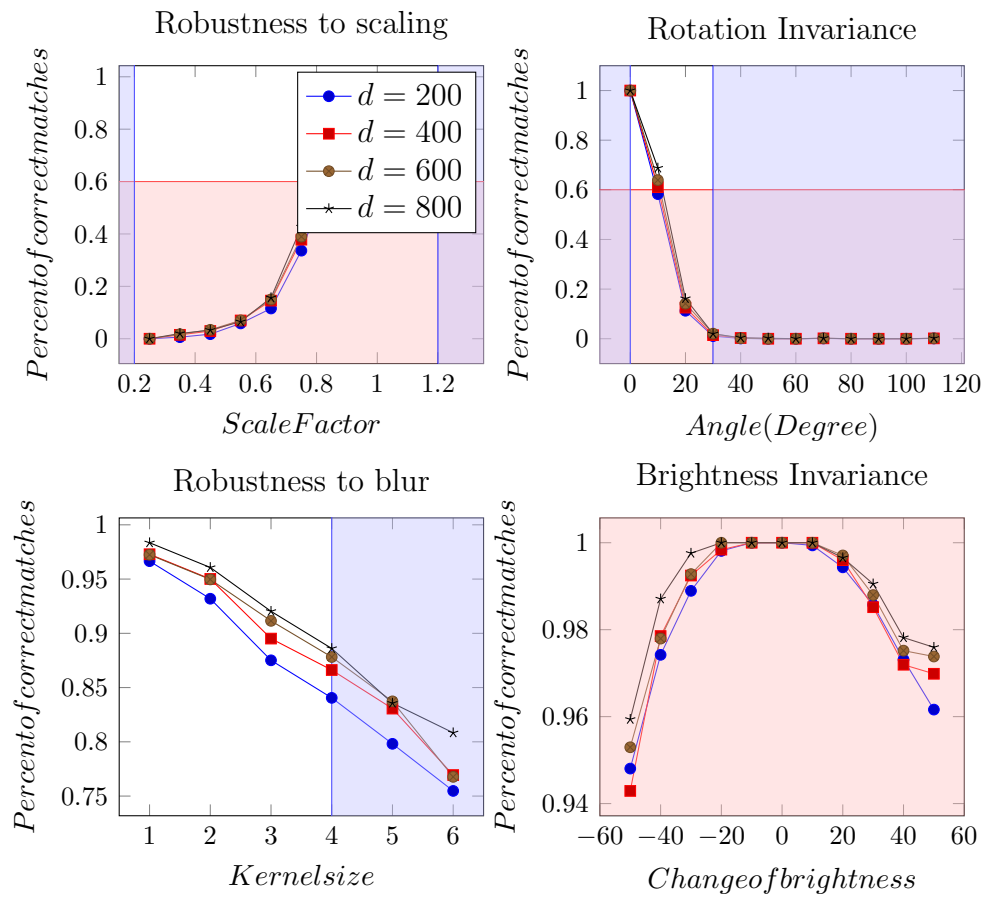


FIGURA 8.15 – Resultado de testes individuais de performance com SURF

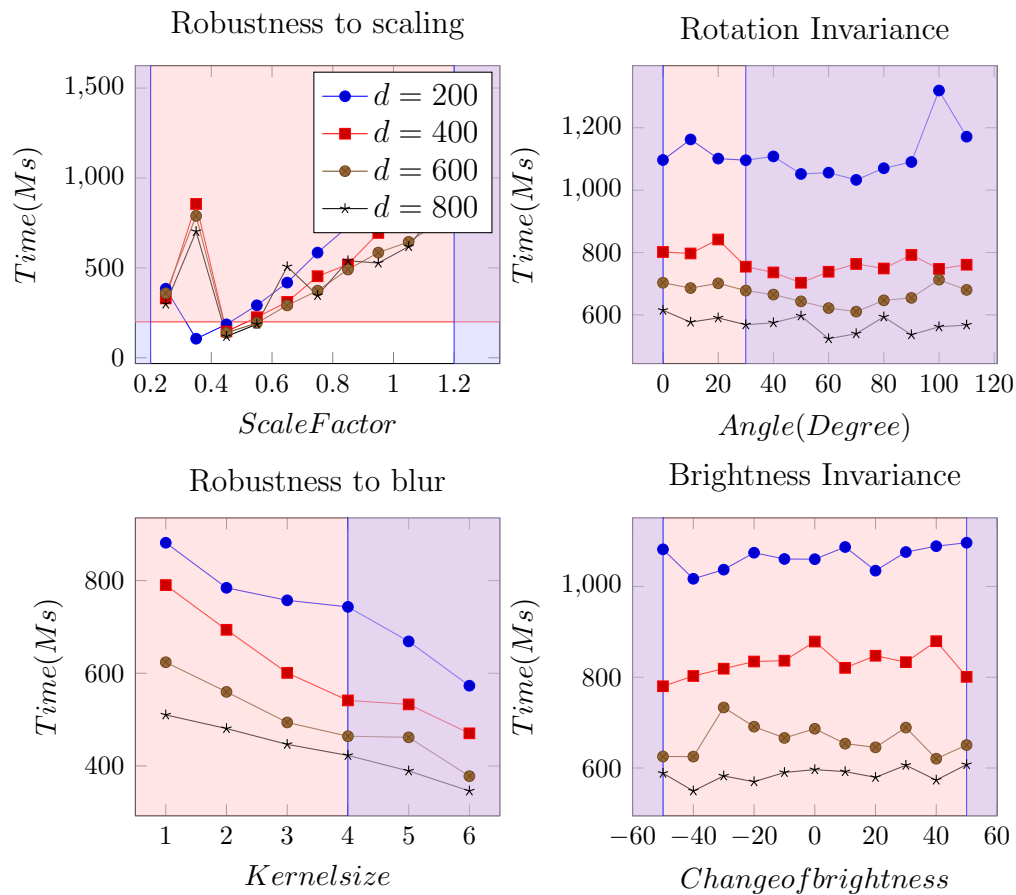


FIGURA 8.16 – Resultado de testes individuais de tempo com SURF

8.2 Análise de resultados

Após a análise pontual de cada algoritmo, podemos compará-los entre si e avaliar o que melhor se adequa ao contexto proposto.

Os algoritmos SIFT e SURF apesar de retornarem bons resultados para o contexto, são pagos e portanto serão desconsiderados para a abordagem proposta.

Uma análise prévia dos algoritmos selecionados tem como tempo de resposta por frame a imagem ?? que nos mostra claramente que a técnica FREAK não responde em tempo ábil para serem considerados tempo real, portanto serão desconsiderados.

A imagem 8.17 que relaciona como as técnicas: BRISK, FAST, FREAK, ORB, STAR, SURF e SIFT respondem a variação

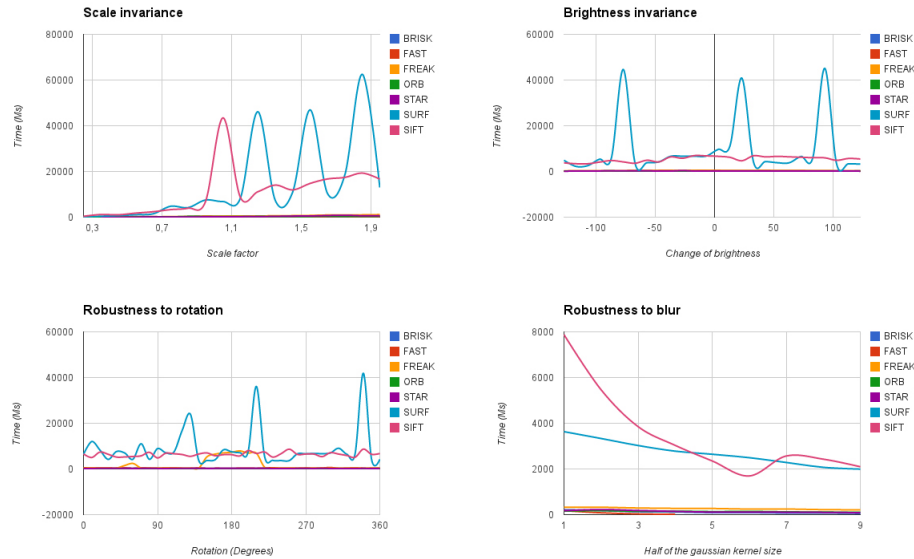


FIGURA 8.17 – Performance quanto à variações de escala, iluminação, rotação e gaussian blur

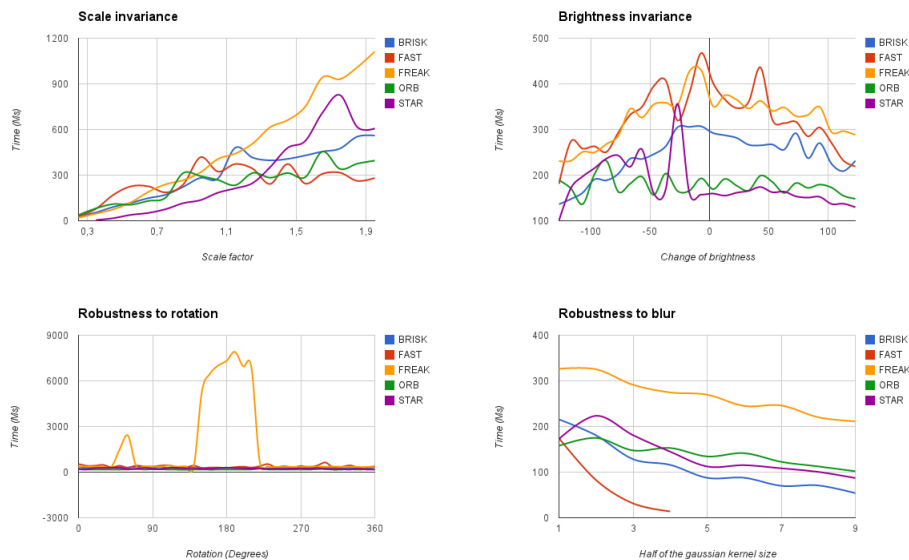


FIGURA 8.18 – Performance quanto à variações de escala, iluminação, rotação e gaussian blur sem os algoritmos SIFT e SURF

8.3 Homografia

Após a etapa de extração de características da imagem padrão e da imagem de comparação é importante fazer o casamento de padrões para que com um número significativo de características, o objeto seja reconhecido. As imagens imagem 8.19, mostram correspondências entre imagens variando as transformações propostas pelo protótipo em comparação com imagens sem as transformações.

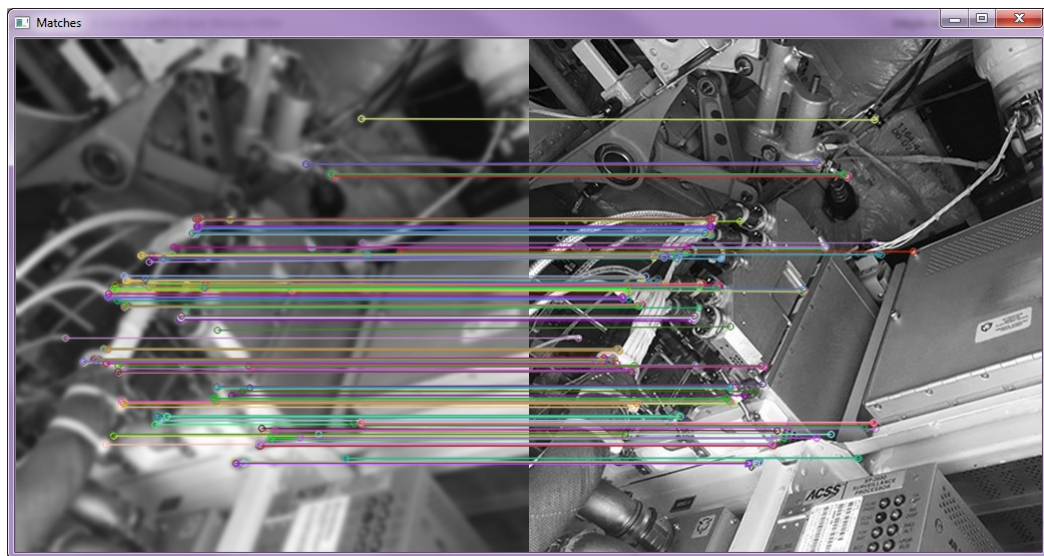


FIGURA 8.19 – Homografia com variação de Blur.

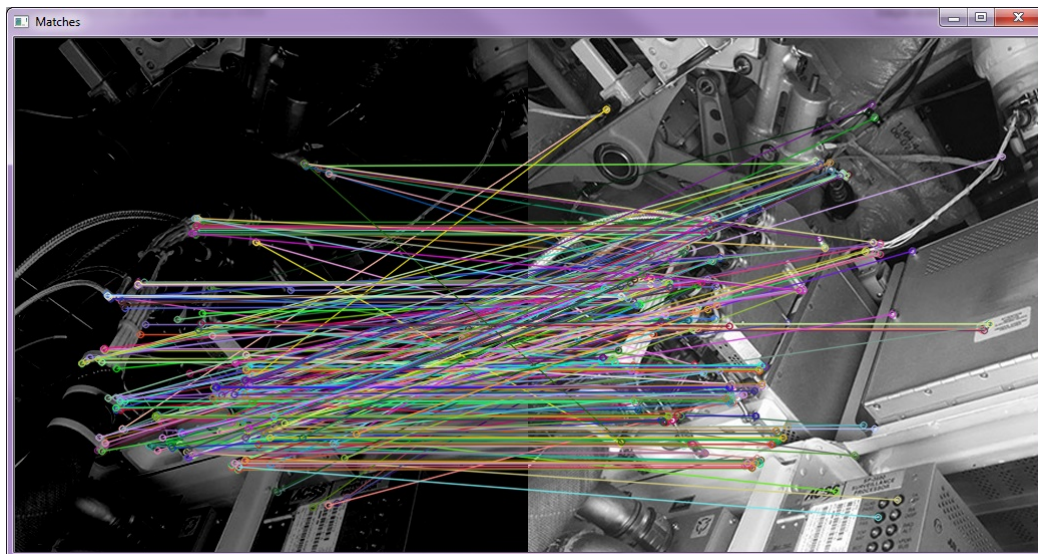


FIGURA 8.20 – Homografia com variação variação de iluminação.

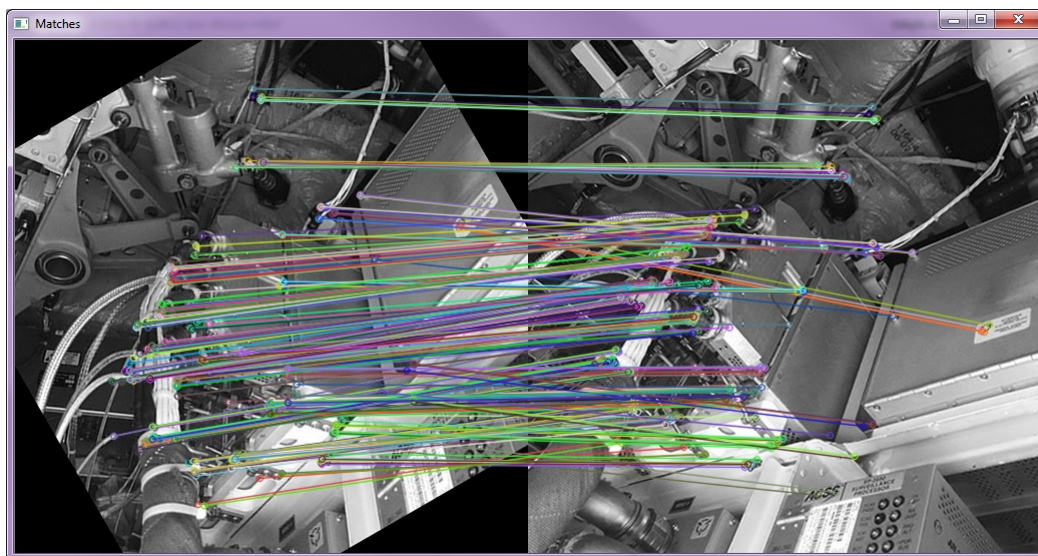


FIGURA 8.21 – Homografia com variação variação de Rotação.

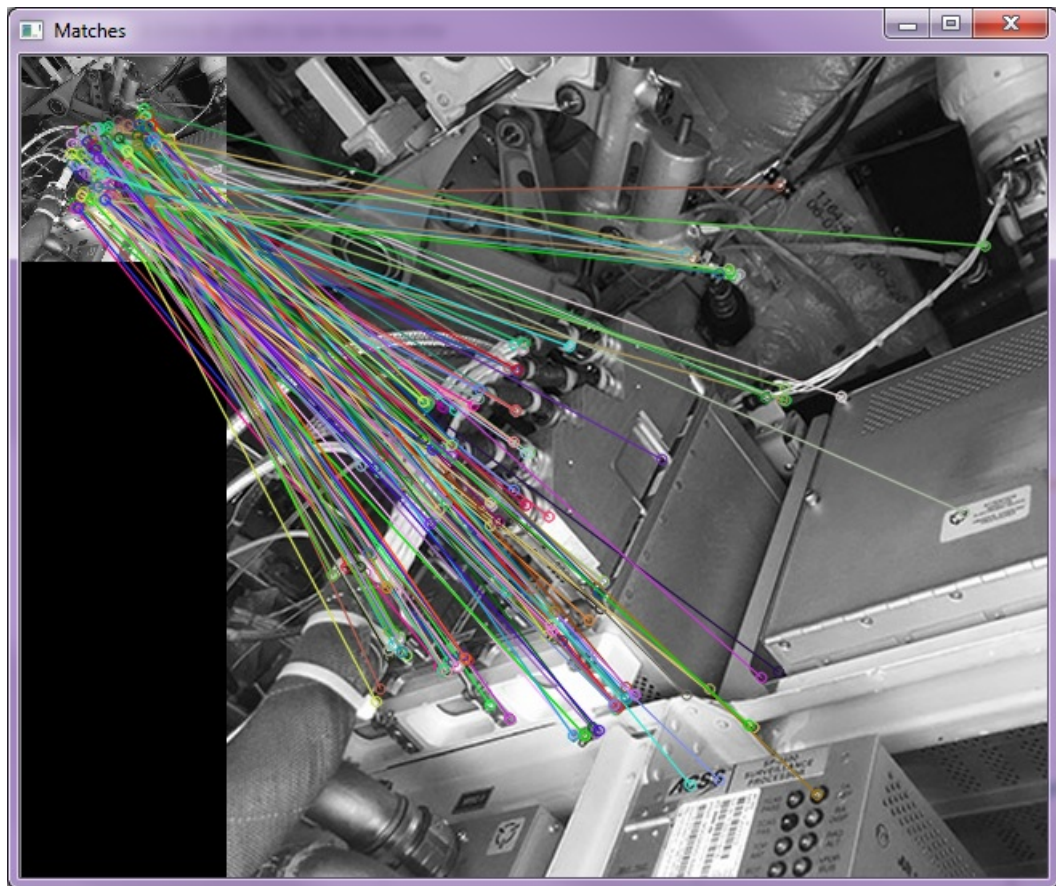


FIGURA 8.22 – Homografia com variação variação de Escala.

([GAY-BELLILE, 2012](#))

9 Conclusão

10 Trabalhos futuros

A utilização de Realidade Aumentada no campo da manutenção pode trazer muitos ganhos no que tange à usabilidade levando ao usuário uma quantidade de informações que da maneira tradicional por meio inspeção e consulta em manuais seria inviável.

Este trabalho teve como foco o reconhecimento de padrões em um cenário aeronáutico específico, como próximos passos temos:

- Adequar a aplicação para dispositivos móveis como tablets, celulares ou mesmo dispositivos HMD de forma a dar mais flexibilidade ao condutor da manutenção;
- Realizar o casamento de padrões com vídeos e imagens tempo real utilizando as técnicas identificadas, otimizando a aplicação para se tornar o mais tempo real e aceitável possível;
- Adaptar a aplicação para utilizar processamento paralelo e processamento em GPU visto os algoritmos serem recursivos e localmente independentes.
- Analisar por meio de testes em campo com possíveis usuários para abstrair

parâmetros de usabilidade como por exemplo determinar que tipo de informações seriam úteis ao usuário ou mesmo que tipo de configuração de dispositivo seria o mais adequado para uma aplicação desse porte.

Referências Bibliográficas

ALEXANDRE ALAHI RAPHAEL ORTIZ, P. V. Freak: Fast retina keypoint. **Ecole Polytechnique Fed´erale de Lausanne (EPFL), Switzerland´**, 2012.

AZUMA, R. T. A survey of augmented reality. **Hughes Research Laboratories**, aug 1997.

E. ROSTEN, G. R.; DRUMMOND, T. Real-time video annotations for augmented reality. **International Symposium on Visual Computing**, 2005.

GARY BRADSKI, A. K. **Learning OpenCv Computer vision with the Opencv Library**. [S.l.]: O´REILLY, 2008.

GAY-BELLILE, V. A mobile markerless augmented reality system for the automotive field. 2012.

HERBERT BAY TINNE TUYTELAARS, L. V. G. Surf : Speeded up robust features. **ETH Zurich**, 2008.

KRYSTIAN MIKOLAJCZYK, C. S. A performance evaluation of local descriptors. feb 2005.

LEUTENEGGER, S.; CHLI, M.; SIEGWART, Y. Brisk: Binary robust invariant scalable keypoints. In: **In Computer Vision (ICCV), 2011 IEEE International Conference on**. [S.l.: s.n.], 2011. p. 2548–2555.

LOWE, D. G. Distinctive image features from scale-invariant keypoints. **Computer Science Department University of British Columbia Vancouver, B.C., Canada**, 2004.

ROSTEN, E.; DRUMMOND, T. Fusing points and lines for high performance tracking. **IEEE International Conference on Computer Vision**, vol. 2, p. pp. 15081515, 2005.

ROSTEN, E.; DRUMMOND, T. Machine learning for high-speed corner detection. In: **In European Conference on Computer Vision**. [S.l.: s.n.], 2006. p. 430–443.

RUBLEE, E. Orb: an efficient alternative to sift or surf. 2011.

SILVA, R.; OLIVEIRA, J. C.; GIRALDI, G. A. **Introduction to augmented reality**. [S.l.], 2003.

TANG, J. C. et al. Why do users like video. In: **Computer Supported Cooperative Work**. [S.l.]: Kluwer Academic Publisher, 1993. p. 163–196.

TINNE TUYTELAARS, K. M. Local invariant feature detectors: A survey. 2007.

FOLHA DE REGISTRO DO DOCUMENTO

1. CLASSIFICAÇÃO/TIPO TM	2. DATA 24 de dezembro de 1969	3. DOCUMENTO Nº CTA/ITA - IEC/TM-002/1969	4. Nº DE PÁGINAS 70
5. TÍTULO E SUBTÍTULO: Avaliação de técnicas de reconhecimento de padrões em ambientes aeronauticos			
6. AUTOR(ES): Bruno Duarte Corrêa			
7. INSTITUIÇÃO(ÕES)/ÓRGÃO(S) INTERNO(S)/DIVISÃO(ÕES): Instituto Tecnológico de Aeronáutica. Divisão de Ciência da Computação – ITA/IEC			
8. PALAVRAS-CHAVE SUGERIDAS PELO AUTOR: Teses; Estilos; Italus			
9. PALAVRAS-CHAVE RESULTANTES DE INDEXAÇÃO: Teses e Dissertações; Estilos; Usuários			
10. APRESENTAÇÃO: ITA, São José dos Campos, 2015, 70 páginas		(X) Nacional () Internacional	
11. RESUMO: O reconhecimento de objetos em uma cena para posterior uso em realidade aumentada depende de diversas variáveis, causando a necessidade do uso de diversas técnicas específicas para cada cenário, sendo portanto, um estudo de fronteiras para a melhor escolha do algoritmo de registro de acordo com a aplicação em questão de grande valia para o meio acadêmico. Esta tese se propõe a estudar, categorizar e traçar fronteiras das técnicas conhecidas tendo como caso de uso manutenção de aeronaves feita dentro de centros fechados.			
12. GRAU DE SIGILO: () OSTENSIVO () RESERVADO () CONFIDENCIAL (X) SECRETO			