**Extended CAMShift Tracking**

**Hugo Rafael Alves da Silva**

Recife, Fevereiro de 2024

**1. Introdução**

Na busca por soluções eficazes em rastreamento de objetos na visão computacional, o algoritmo CAMShift se destaca como uma ferramenta poderosa e versátil. Originado como uma extensão do algoritmo Mean Shift, o CAMShift incorpora estratégias adaptativas que o capacitam a enfrentar os desafios dinâmicos encontrados em ambientes reais. Sua capacidade de ajustar-se continuamente às variações de iluminação, escala e oclusão o torna uma escolha preferida em uma variedade de aplicações, desde vigilância até realidade aumentada.

O CAMShift opera no princípio fundamental de deslocamento iterativo de uma janela no espaço de características para localizar o modo de uma função de densidade de probabilidade. Esta abordagem adaptativa permite que o algoritmo ajuste sua escala e orientação dinamicamente com base nas características do objeto, facilitando o rastreamento preciso e robusto em cenários desafiadores. Ao integrar informações de cor e distribuição espacial, o CAMShift constrói um modelo multidimensional que captura a complexidade das aparências dos objetos, proporcionando uma base sólida para o rastreamento eficiente em tempo real.

A implementação prática do CAMShift envolve uma série de etapas, incluindo extração de características, computação de histogramas, e deslocamento médio iterativo. Esses procedimentos, combinados com a capacidade adaptativa do algoritmo, conferem-lhe uma ampla gama de aplicações, desde monitoramento de tráfego até interação humano-computador. Além disso, o CAMShift continua a ser objeto de estudo e aprimoramento, com pesquisas em andamento visando otimizar sua robustez e escalabilidade para atender às demandas crescentes da visão computacional em tempo real.

No entanto, apesar de suas vantagens notáveis, o CAMShift não está isento de limitações. Uma das principais falhas do algoritmo reside em sua sensibilidade a mudanças abruptas na aparência do objeto em cena, especialmente em situações de oclusão ou alterações drásticas na iluminação. Nessas circunstâncias, o CAMShift pode perder temporariamente o rastreamento do objeto-alvo, resultando em imprecisões ou falhas completas. Além disso, a capacidade de adaptação do CAMShift pode ser comprometida em ambientes altamente dinâmicos, nos quais a aparência do objeto muda drasticamente ao longo do tempo. Estas limitações destacam a necessidade contínua de pesquisa e desenvolvimento de técnicas complementares ou aprimoradas, a fim de mitigar as deficiências do CAMShift e melhorar sua robustez em uma ampla gama de cenários de aplicação.

Para superar essas limitações, fora proposta uma extensão ao algoritmo CAMShift, chamada de Extended CAMShift Tracker. Essa extensão, desenvolvidas com baixo custo computacional, visa aprimorar a robustez e a eficácia do rastreamento de objetos em várias condições desafiadoras. Uma das principais contribuições do Extended CAMShift Tracker é a capacidade de acumular múltiplos histogramas para modelar as diferentes aparências dos objetos-alvo. Essa abordagem permite uma representação mais abrangente das características visuais do objeto, tornando o rastreamento mais resistente a variações na aparência. Além disso, o algoritmo incorpora técnicas avançadas de identificação de objetos e re-detecção após a perda de trilhas. Isso é alcançado por meio da comparação contínua dos histogramas dos objetos rastreados com referências previamente armazenadas e uma estratégia de re-detecção hierárquica para recuperar objetos perdidos.

O restante desse artigo é divido nas seguintes seções: Seção 2 contendo a metodologia do estudo, mostrando a implementação do CAMShift e do Extended CAMShift Tracker; Seção 3 contendo uma síntese dos resultados obtidos pelos algoritmos; e a Seção 4 contendo a discussão e a conclusão.

**2. Metodologia**

Esse projeto foi desenvolvido a partir de um estudo, desenvolvido por David Exner, intitulado Fast and Robust CAMShift Tracking. A partir disso, coube o estudo da ferramenta CAMShift e das técnicas usadas por Exner para a criação do Extended CAMShift. O artigo base não possuía código disponibiliza código, logo o objetivo desse projeto é implementar esse código de acordo com os pontos observados no artigo.

**2.1 CAMShift**

Para o algoritmo do CAMShift utilizei a implementação disponibilizada em sua documentação, já que este irá servir como base para o Extended CAMShift Tracker. O código em questão faz uma aplicação básica do CAMShift em um vídeo de um tráfego, desenhando uma janela (calculada pela ROI – Return Of Investment) que tenta acompanhar um determinado veículo no vídeo. O código do CAMShift está no repositório anexo a este documento.

**2.2 Extended CAMShift Tracker**

O algoritmo proposto por Exner possui três melhorias em cima do CAMShift comum. Essas melhorias foram implementadas para suprir falhas do algoritmo base assim como complementar uma a outra, já que as próprias técnicas, sozinhas, também possuem limitações relacionadas à oclusão ou perca de rastreamento.

*2.2.1 Acumulação de múltiplos histogramas*

O rastreamento de objetos com aparências complexas pode representar um desafio para o algoritmo padrão CAMShift. Para superar essa limitação, uma abordagem proposta é a utilização de múltiplos histogramas para representar diferentes aparências dos objetos-alvo. Durante a execução do algoritmo, os histogramas acumulados são empregados para calcular a retroprojeção de probabilidade, enquanto os histogramas individuais auxiliam na identificação dos objetos. Essa abordagem proporciona uma representação compacta e abrangente das múltiplas visões de aparência dos objetos, sem acarretar um custo significativo de desempenho.

*2.2.2 Identificação de objetos*

O rastreamento de múltiplos objetos similares pode se tornar desafiador para o CAMShift padrão, especialmente quando esses objetos interagem ou se sobrepõem. Para resolver essa questão, é crucial monitorar continuamente a identidade de cada objeto-alvo. Isso é realizado por meio da comparação contínua do histograma de cada objeto com histogramas de referência previamente armazenados. Caso a correspondência entre os histogramas caia abaixo de um determinado limite, indicando uma potencial oclusão ou perda, é iniciado um processo de re-detecção.

*2.2.3 Redetecção hierárquica*

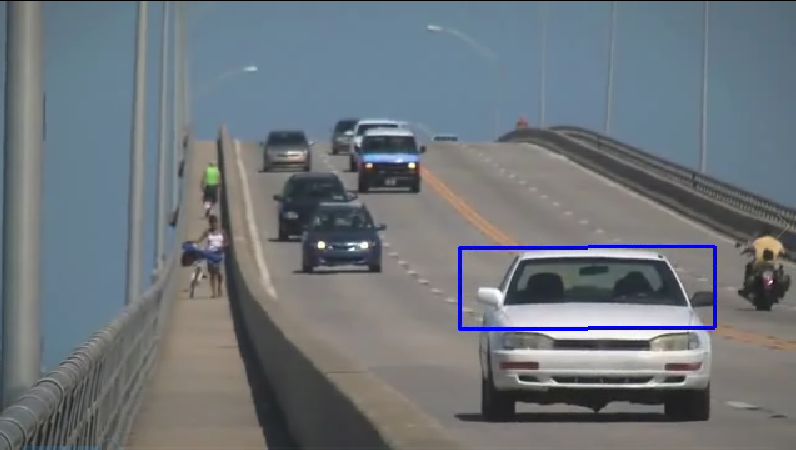
Objetos podem ser completamente ocultados ou temporariamente sair do campo de visão da câmera, o que requer um método confiável de re-detecção. Métodos tradicionais, como o CAMShift padrão, podem enfrentar dificuldades na re-detecção, especialmente na presença de objetos com cores semelhantes. Uma solução proposta é a utilização de uma estratégia hierárquica de re-detecção com base em árvores quaternárias. Nessa abordagem, a janela de busca é subdividida recursivamente em quadrantes, e o rastreador CAMShift estendido é aplicado a cada subdivisão até que o objeto perdido seja re-detectado com sucesso.

**3. Resultados**

As três melhorias propostas por Exner foram sintetizadas no código base do CAMShift (contido em repositório anexo a este documento). Após isso, executamos os dois algoritmos para notar diferenças em eficiência.

O CAMShift simples acompanha a movimentação do veículo até que este saia do percurso e por fim acaba se perdendo, não identificando outros objetos similares que se movam.

CAMShift base acompanhando o percurso do objeto

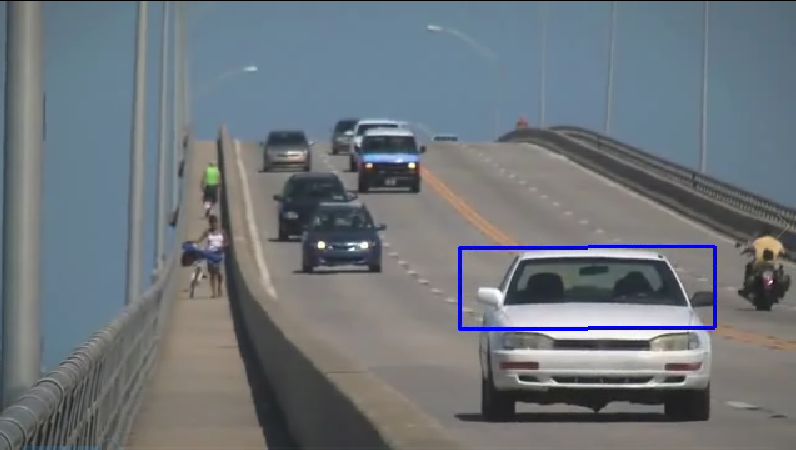


CAMShift base falhando em acompanhar a movimentação de objetos similares



Já o Extended CAMShift Tracker ele não comete o mesmo erro do CAMShift base, pois recupera a detecção de movimentação de outros objetos similares.

Extended CAMShift Tracker acompanhando o percurso do objeto



Extended CAMShift Tracker adaptando a janela de busca para encontrar outros objetos similares em movimento



Os resultados obtidos pelo Extended CAMShift Tracker também estão disponíveis no repositório anexo a este documento, contendo cada vídeo para a comparação dos resultados.

**4. Conclusão**

O CAMShift é um algoritmo com diversas aplicações e utilidades, ainda que apresente algumas falhas para determinadas aplicações. O trabalho de Exner propõe uma melhoria nesse algoritmo para que ele se torne mais robusto e rápido. Nesse projeto, aplicando as três melhorias de Exner ao CAMShift, pudemos notar que há uma melhoria significativa do algoritmo para o problema proposto. Contudo, por limitações de hardware, nem todas as melhorias propostas puderam ser testadas, já que Exner direciona sua solução para GPU.

O Extended CAMShift Tracker se mostra satisfatório em uma primeira análise de seus resultados sem um código tão bem elaborado. Essa falta de especificação do desenvolvimento das melhorias propostas por Exner para o CAMShift podem causar limitações na replicação do projeto, dificultando sua avaliação.

Por fim, caberia como trabalho futuro especificar os pontos de melhoria de Exner para que o código pudesse ser mais bem elaborado e, consequentemente, bem aplicado aos problemas propostos, ao invés de problemas específicos.