Otimizando a Transformada de Hough com Interseção de Semiespaços

João Felipe Santos
Escola de Engenharia
Universidade Federal Fluminense
Niterói, Brasil
Email: santos.joaofelipe@gmail.com

Leandro A. F. Fernandes
Instituto de Computação
Universidade Federal Fluminense
Niterói, Brasil
Email: laffernandes@ic.uff.br

***Abstract*—A recurrent problem in computer vision is identifying multiple types of geometrical patterns in datasets. This way, techniques that reduce the computational cost and memory footprint of automatic recognition processes are extremely important. This work proposes a solution for significantly reducing the memory allocation needs that are a common drawback of Hough Transform-based geometrical pattern recognition algorithms.**

***Keywords*-Transformada de Hough; Semiespaços**

I. Introdução

A Transformada de Hough (TH) é uma técnica bastante conhecida e de grande uso devido a sua simplicidade conceitual, capacidade de retornar todas as estruturas com a organização geométrica procurada (e.g., retas, círculos, elipses) e de robustez sobre dados ruidosos. Contudo, ela requer grande custo computacional e aloca grande quantidade de memória na representação do mapa de votos em seu processamento. Além disso, o equacionamento da TH precisa ser especializado para o tipo de estrutura geométrica desejada, o que faz com que todos os métodos atuais para tentar otimizá-la tem se limitado ao caso específico do tipo de geometria por ela contemplado (e.g., otimizações desenvolvidas para detecção de retas a partir de pontos não estendem de maneira direta para detecção de círculos a partir de pontos e informação do gradiente calculado no ponto). Mais informações sobre a TH, sua importância e o mapa de votos podem ser encontrados em [1].

Foi definido por Fernandes e Oliveira [1] um processo mais geral que utiliza álgebra geométrica na generalização da TH para qualquer forma analítica em qualquer dimensionalidade sem a necessidade de se especializar seu equacionamento. Apesar das questões relacionadas a custo computacional e processamento permanecerem neste método, ele permite que as melhorias desenvolvidas generalizem de de maneira direta todos os casos de aplicação.

O presente trabalho se propõe a otimizar a generalização da TH apresentada em Fernandes e Oliveira [1] por meio do uso de estruturas de dados sofisticadas para gerência do mapa de votos, eliminando, assim, a necessidade de um mapa de votos discreto representado como uma grande quantidade de memória alocada. Visando otimizar estre processo, representamos os objetos mapeados para o espaço de parâmetros contínuo, como flats (i.e., estrutura planares tais como ponto, reta, plano e seus equivalentes em dimensões mais altas) discretos. Com isso, transferimos a discretização do mapa de votos para os elementos mapeados. Cada flat discreto, por sua vez, é representado por um conjunto de pares de semiespaços (i.e., pares de inequações que definem restrições lineares), e a partir disto, utilizando uma estrutura de dados baseada na árvore BSP (*binary space partitioning*), avaliamos as possíveis interseções entre os semiespaços para encontrar as regiões do espaço de parâmetros onde os votos são acumulados.

** João Felipe Santos é bolsista IC-FAPERJ, Processo E-26/102.047/2011.

II. Background Técnico

Esta seção comenta as ferramentas matemáticas necessárias para o desenvolvimento de nossa técnica.

A. Standard Model

Estando num espaço n-dimensional, um hiperplano discreto (i.e., flat discreto (n-1)-dimensional) pode ser descrito como a região de interseção de um par de hiperplanos que distam de meia unidade da equação que define o hiperplano contínuo original. Para flats de dimensionalidades menores, o mesmo processo é feito, não no espaço n-dimensional, e sim em suas projeções em dimensionalidades mais baixas até que o flat projetado seja um hiperplano desta dimensionalidade.

A partir disso, dado um flat de dimensionalidade qualquer, é possível defini-lo como uma coleção de pares de semiespaços, que definem restrições. A Figura 1 ilustra o caso de uma reta em 3D, que não é um hiperplano de sua atual dimensionalidade, e de suas projeções em 2D nas faces do cubo. A Figura 2 demonstra a discretização de uma projeção em uma das faces do cubo da Figura 1. Por restrições de espaço, não foi possível expor seu equacionamento, mas este pode ser encontrado no artigo de Andres [2].

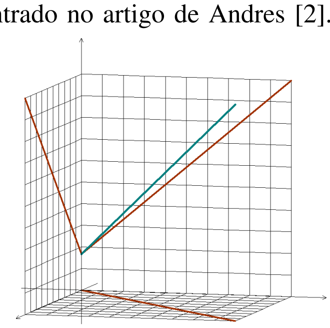

Fig. 1. Em verde, reta num espaço tridimensional e em marrom, suas projeções em dimensionalidades menores, onde ela será um hiperplano

Andres [2] apresenta o Standard Model como um processo sistemático para a discretização de objetos euclidianos, dentre eles flats, em um espaço afim por meio de projeções e definições de restrições. Neste trabalho nós utilizamos o Standard Model para discretizar os dados de entrada da TH que foram mapeados para o espaço de parâmetros contínuo.

B. Árvore de votação

Uma árvore BSP é uma estrutura de dados que divide recursivamente o espaço, seja ele de qualquer dimensionalidade, com hiperplanos particionadores definidos a partir dos elementos que propôe a organizar.

Neste trabalho desenvolvemos uma variação da árvore BSP para organizar as restrições geradas por cada flat discreto e, a cada nodo da árvore, tratamos a interseção dessas restrições e o acúmulo dos votos da transformada generalizada.

C. Poliedros de intersecção

Ao aplicar o processo definido por Avis e Fukuda [3] para encontrar os vértices de regiões definidas pelas interseções de semiespaços, nós encontramos os pontos de interseção entre as fronteiras das restrições. Estes pontos definem vértices de poliedros convexos onde efetuaremos o novo tipo de processo de votação proposto. Um poliedro delimitado pelo tipo de vértices acima descrito é basicamente a região do espaço onde uma ou mais interseções de objetos discretos acontecem.

III. Novo esquema de votação proposto

O Standard Model (Seção II-A) é definido para ser utilizado em um espaço afim. Contudo, o espaço de parâmetros da técnica apresentada por Fernandes e Oliveira [1] mapeia dados de entrada para superfícies curvas em um espaço de parâmetros hiperesférico. Felizmente, existe uma relação entre o espaço de sua transformada, e a cobertura afim aberta da grasmanniana, o que permite utilizar o Standard Model na discretização das estruturas envolvidas no processo de votação.

A partir das restrições (Seção II-A), é usada a árvore de votação (Seção II-B) para organizar os objetos e durante este processo avaliar os pontos onde ocorrem interseções entre as fronteiras dos semiespaços (Seção II-C). Estes pontos definem os vértices de poliedros de interseções, que correspondem à região do espaço de parâmetros onde votos são acumulados.

Após o término da votação, os poliedros com mais votos são retornados como as representantes das estruturas geométricas mais aparentes no conjunto de dados de entrada. Esta etapa final é equivalente à busca por picos de votos no mapa de acumuladores da TH.

IV. Implementação

Na implementação da técnica proposta nós utilizamos a linguagem de programação C++. Também foi usada a biblioteca Boost, mais especificamente a uBlas, para tratamento de operações com álgebra linear.

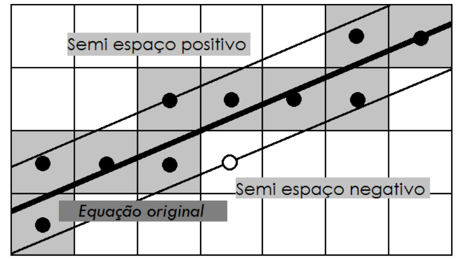

Fig. 2. Exemplo do Standard Model num espaço discreto, criando o par de restrições que distam meia unidade da equação original

V. Experimentos

Por ser um trabalho em andamento, não é possível verificar com precisão até que ponto a melhoria proposta contribui com a otimização da TH generalizada de Fernandes e Oliveira [1]. No estágio atual do projeto é possível fazer testes unitários em cada passo implementado. O Standard Model e a árvore de votação já foram devidamente implementados e validados. Atualmente estamos na fase de obtenção dos vértices dos poliedros de interseção.

VI. Resultados e discussões

Diferente do método convencional para a contagem de votos, onde para uma dada dimensionalidade tem-se um número fixo de elementos a serem alocados para qualquer situação, no novo esquema de votação teremos uma complexidade de armazenamento crescente à medida que a quantidade de dados de entrada aumenta. Porém, mesmo assim, nossas estimativas mostram que seria necessário um cenário extremamente sobrecarregado para se chegar à quantidade de memória alocada pelo método convencional.

Comparando a melhoria proposta e o método que é atualmente empregado, tendo um espaço de parâmetros contínuos tridimensional de 100x100x100, tomando um passo de discretização de 5 unidades, por exemplo, teríamos 20^3 voxels e precisaríamos armazenar o mesmo número de pontos flutuantes para caracterizar o espaço onde seriam contabilizados os votos. Em contrapartida, tomando um caso usual como a interseção de dois planos, no novo método, precisaríamos de 28 pontos flutuantes para definir a região do espaço para efetuar a contagem de votos.

VII. Conclusão

Neste trabalho nós apresentamos uma técnica para otimização de alocação de memória do mapa de votos para um espaço de parâmetros contínuo da TH generalizada para formas analíticas. A partir disso esperamos obter grandes melhorias em termos de quantidade de memória e tempo de processamento necessário para execução da TH.

Referências

[1] L. A. F. Fernandes and M. M. Oliveira, “A general framework for subspace detection in unordered multidimensional data”, Pattern Recognition, vol. 45, pp. 3566-3579, 2012.

[2] E. Andres, “Discrete linear objects in dimension n: the standard model”, Graphical Models, vol. 65, pp. 92–111, 2003.

[3] D. Avis and K. Fukuda, “A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra”, Discrete & Computational Geometry, vol. 8, pp. 295–313, 1992.