

Nội dung chính

1 Giới thiệu bài toán

- Đặt vấn đề
- Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)

2 Thuật toán tính bao lồi xấp xỉ

- Outer convex approximation

3 Thuật toán tính bao lồi trực giao

- Các khái niệm liên quan
- Thuật toán tìm bao lồi - Quickhull

4 Thực nghiệm và kết quả

- Môi trường

Nội dung chính

1 Giới thiệu bài toán

- Đặt vấn đề
- Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)

2 Thuật toán tính bao lồi xấp xỉ

- Outer convex approximation

3 Thuật toán tính bao lồi trực giao

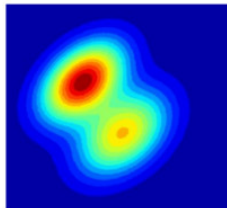
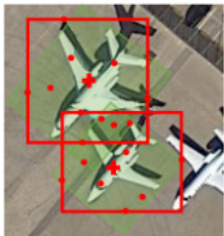
- Các khái niệm liên quan
- Thuật toán tìm bao lồi - Quickhull

4 Thực nghiệm và kết quả

- Môi trường

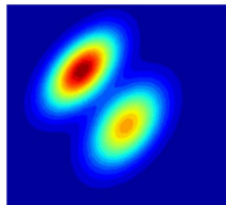
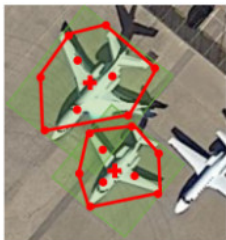
Đặt vấn đề

Thách thức lớn: đặc trưng răng cưa (feature aliasing)



Giải quyết vấn đề

Sử dụng bao lồi làm biểu diễn bounding box.



Nội dung chính

1 Giới thiệu bài toán

- Đặt vấn đề
- Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)

2 Thuật toán tính bao lồi xấp xỉ

- Outer convex approximation

3 Thuật toán tính bao lồi trực giao

- Các khái niệm liên quan
- Thuật toán tìm bao lồi - Quickhull

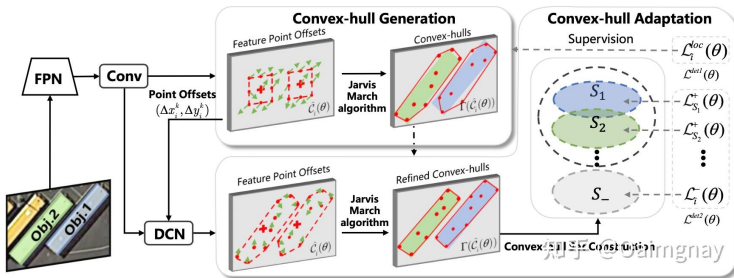
4 Thực nghiệm và kết quả

- Môi trường

Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)

Giới hạn phạm vi đối tượng sử dụng chỉ số CloU.

Phân chia tập bao lồi thành bao lồi âm và bao lồi dương.



Xây dựng tập bao lồi

Phương pháp CFA đã đề xuất biểu diễn phạm vi của đối tượng bằng bao lồi:

$$C_i = \{(x_i^k, y_i^k)\}_i^{k=1,2,\dots,K} \quad (1)$$

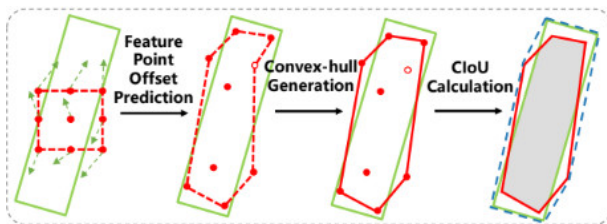
2 giai đoạn thực hiện:

- I. Tạo và ước lượng bố cục bao lỗi.
- II. Chỉnh sửa bao lỗi để phù hợp với các đối tượng dày đặc.

Giai đoạn I

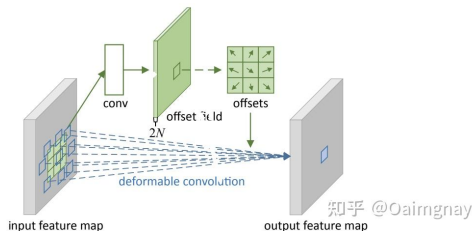
Dự đoán độ lệch:

$$\hat{C}_l(\theta) \leftarrow \{(x_i^k + \Delta x_i^k, y_i^k + \Delta y_i^k)\}_{i=1,2,\dots,K}^k \quad (2)$$



Tích chập biến dạng

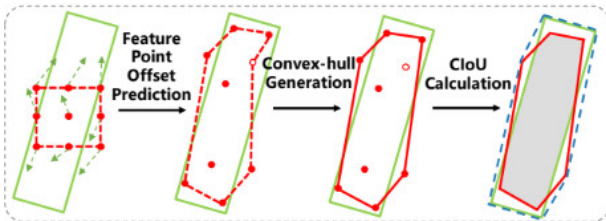
Hỗ trợ lấy mẫu ở những vị trí đa dạng hơn => làm giàu dữ liệu.



Thuật toán tìm bao lồi

- Thuật toán tìm bao lồi xấp xỉ - Convex Approximation.
- Thuật toán tìm bao lồi trực giao - Orthogonal Quick Hull.
Tìm một bao lồi tối thiểu cho 9 điểm feature point offset.

Định nghĩa công thức Convex Intersection over Union (CIoU)



$$CIoU_{(C_i(\theta), B_j)}(\theta) = \frac{|C_i(\theta) \cap B_j|}{|C_i(\theta) \cup B_j|} - \frac{|R_j \setminus (C_i(\theta) \cup B_j)|}{|R_j|} \quad (3)$$

Convex-Hull Localization Loss

$$\mathcal{L}_i^{loc}(\theta) = 1 - CloU(C_i(\theta), B_j) \quad (4)$$

Convex-hull Classification Loss

$$\mathcal{L}_i^{cls}(\theta) = FL(S_i(\theta), Y_j) \quad (5)$$

Convex-hull Loss

Hàm loss phân loại cho bao lỗi dương:

$$\mathcal{L}^+(\theta) = \mathcal{L}_i^{cls}(S_i(\theta), Y_j) + \lambda \mathcal{L}_i^{loc}(C_i(\theta), B_j) \quad (6)$$

Hàm loss phân loại cho bao lỗi âm:

$$\mathcal{L}^-(\theta) = \mathcal{L}_i^{cls}(S_i(\theta), Y_j) \quad (7)$$

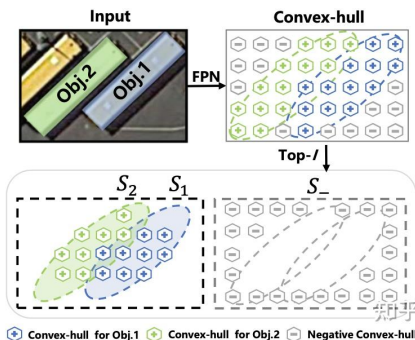
Tối ưu quá trình huấn luyện

$$\mathcal{L}^{\text{det } 1}(\theta) = \frac{1}{J} \sum_i \mathbb{I}_{(x_i, y_i)} \mathcal{L}_i^{\text{loc}}(\theta) \quad (8)$$

Thích ứng bao lồi - Convex Hull Adaptation

Xử lý hiện tượng feature aliasing.

Convex-Hull Set Construction: Xây dựng một tập các bao lồi cho mỗi đối tượng.



知乎 @Oairngnay

Xây dựng tập các bao lỗi

Tập các bao lỗi dương(S_j) được xây dựng bằng cách chọn ra top-l bao lỗi làm bao lỗi dương, theo CloU giữa các bao lỗi và các hộp thật của đối tượng (growth-truth).

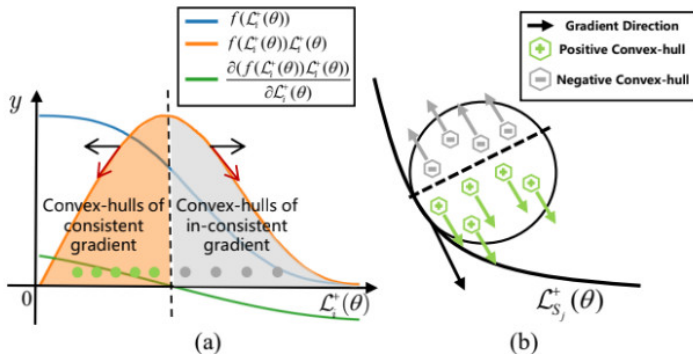
Các bao lỗi khác mà không thuộc đối tượng nào sẽ được gộp vào tập các bao lỗi âm (S_-).

Chiến lược chia tập các bao lồi

Cách chia tập các bao lồi được hướng dẫn bởi nguyên tắc nhất quán đạo hàm. Lấy đạo hàm của phương trình(??), có đạo hàm của tập các bao lồi:

$$\frac{\partial \mathcal{L}_{S_j}^+(\theta)}{\partial(\theta)} = \frac{1}{|S_j|} \sum_{i \in S_j} \frac{\partial(f(\mathcal{L}_i^+(\theta))\mathcal{L}_i^+(\theta))}{\partial(\mathcal{L}_i^+(\theta))} \frac{\partial \mathcal{L}_i^+(\theta)}{\partial(\theta)} \quad (9)$$

Chiến lược phân đoạn tập các bao lồi



Hình: Chia tách tập convex-hull dựa trên nguyên tắc tính thống nhất gradient.

Xử lý hiện tượng đặc trưng răng cưa

Đưa ra công thức tính hệ số khử đặc trưng răng cưa:

$$p_i = \gamma \frac{CloU(C_i, B_j)}{\sum_{m=1}^M CloU(C_i, B_m)} \quad (10)$$

Tối ưu hàm loss giai đoạn 2

Việc tối ưu của giai đoạn 2 được điều khiển bởi sự kết hợp cả hàm loss classification và localization định nghĩa trên tập các bao lỗi:

$$\mathcal{L}^{det2}(\theta) = \frac{1}{J} \sum_{j=1}^J \frac{1}{|S_j|} \sum_{i \in S_j} p_i f(\mathcal{L}_i^+(\theta)) \mathcal{L}_i^+(\theta) + \frac{1}{|S_-|} \sum_{i \in S_-} \mathcal{L}_i^-(\theta) \quad (11)$$

Hàm loss của bộ phát hiện CFA

Là tổng hàm loss của cả hai giai đoạn:

$$\mathcal{L}_{CFA} = \mathcal{L}^{\text{det}1}(\theta) + \mathcal{L}^{\text{det}2}(\theta) \quad (12)$$

Nội dung chính

1 Giới thiệu bài toán

- Đặt vấn đề
- Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)

2 Thuật toán tính bao lồi xấp xỉ

- Outer convex approximation

3 Thuật toán tính bao lồi trực giao

- Các khái niệm liên quan
- Thuật toán tìm bao lồi - Quickhull

4 Thực nghiệm và kết quả

- Môi trường

Nội dung chính

1 Giới thiệu bài toán

- Đặt vấn đề
- Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)

2 Thuật toán tính bao lồi xấp xỉ

- Outer convex approximation

3 Thuật toán tính bao lồi trực giao

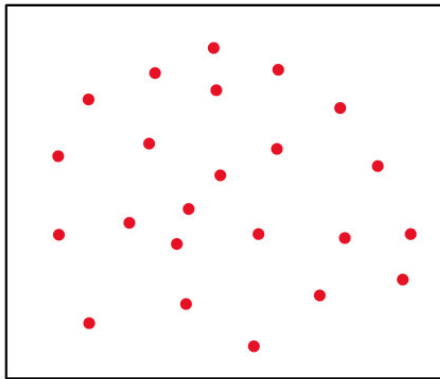
- Các khái niệm liên quan
- Thuật toán tìm bao lồi - Quickhull

4 Thực nghiệm và kết quả

- Môi trường

Thuật toán 1

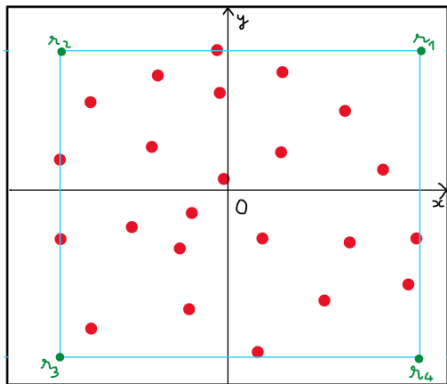
Cho tập hợp các điểm trong không gian:



Cần tìm ra tứ giác bao quanh xấp xỉ các điểm này.

Bắt đầu thuật toán

Khởi tạo hình chữ nhật lớn nhất bao quanh các điểm.



Hình chữ nhật \mathcal{P}^{outer} cấu tạo gồm 4 đỉnh như sau: Tập P ban đầu chứa 4 đỉnh này:

$$P := \{r_1, r_2, r_3, r_4\} \quad (13)$$

Lấy 1 đỉnh $p \in P$, có được p^- là điểm liền trước (ngược chiều kim đồng hồ) của p , p^+ là điểm liền sau của p , ta tính được hướng d_p của p và ngưỡng β_{dp} :

$$\begin{aligned} d_p^T &:= \|p^+ - p^-\|^{-1} R (p^+ - p^-)^T, \\ \beta_{dp} &:= \max\{d_p x^T \mid x \in X\}, \end{aligned} \quad (14)$$

trong đó:

$$R := \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad (15)$$

Vì R là ma trận xoay, thay vào công thức ngưỡng d_p bên trên:

$$\|d_p\| = \|p^+ - p^-\|^{-1} \|(p^+ - p^-)R^T\| = \|p^+ - p^-\|^{-1} \|p^+ - p^-\| = 1. \quad (16)$$

Xét biểu thức định nghĩa \mathcal{P}^{outer} :

$$d_p x^T \leq \beta_{d_p}. \quad (17)$$

Trường hợp 1, nếu:

$$\beta_{d_p} = d_p p^+ \quad (18)$$

Thì ràng buộc (17) sẽ không tạo đỉnh mới mà tạo thêm cạnh mới $[p^-, p^+]$ của \mathcal{P}^{outer} .

Cho $d_{[p^-, p]}$ và $d_{[p, p^+]}$ là hai hướng cực đại từ D định nghĩa hai cạnh $[p^-, p]$ và $[p, p^+]$ của đa giác \mathcal{P}^{outer} . Tuy nhiên hai cạnh này mặc định có sẵn \Rightarrow hai hướng trên trở nên thừa thãi.

Chính vì thế nên khi thêm d_p vào tập D ta cần loại bỏ hai hướng $d_{[p^-,p]}$ và $d_{[p,p^+]}$ khỏi tập D và đỉnh p từ P :

$$\begin{aligned} D &:= (D \cup \{d_p\}) \setminus \{d_{[p^-,p]}, d_{[p,p^+]}\}, \\ P &:= P \setminus \{p\}. \end{aligned} \tag{19}$$

Xét trường hợp 2, nếu:

$$\beta_{d_p} > d_p p^+ \quad (20)$$

và:

$$d_p p^T - \beta_{d_p} > \delta \quad (21)$$

thì ràng buộc (17) tạo thêm 2 đỉnh mới cho đa giác là \hat{p}^- và \hat{p}^+ và được tính toán bởi:

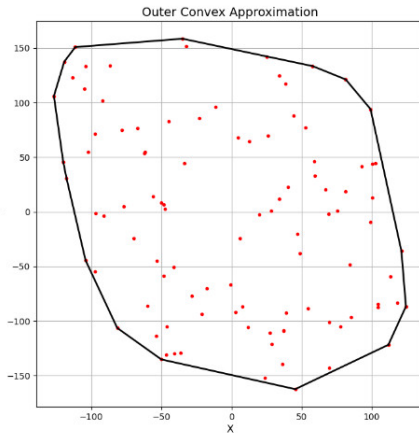
$$\begin{aligned} \lambda_p &:= (\beta_{d_p} - d_p p^{-T}) / (d_p p^T - d_p p^{-T}) \in (0, 1), \\ \hat{p}^- &:= (1 - \lambda_p) p^{-T} + \lambda_p p^T, \\ \hat{p}^+ &:= (1 - \lambda_p) p^{+T} + \lambda_p p^T. \end{aligned} \quad (22)$$

Tiếp theo ta thêm d_p vào D và thay thế $p \in P$ bởi \hat{p}^- và \hat{p}^+ :

$$\begin{aligned} D &:= D \cup \{d_p\}, \\ P &:= (P \setminus \{p\}) \cup \{\hat{p}^-, \hat{p}^+\}. \end{aligned} \tag{23}$$

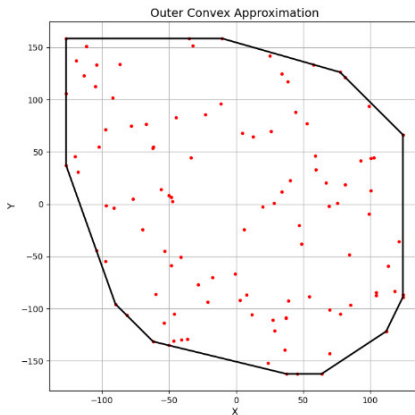
Lặp lại các bước tương tự, ta sẽ tìm được một bao lồi xấp xỉ của tập hợp các điểm.

Một vài kết quả của Outer Convex Approximation



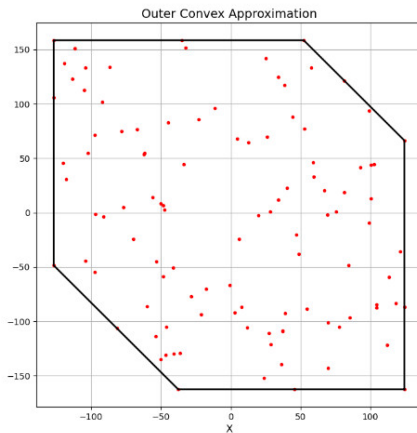
Hình: Với $n = 100$, $\delta = 0$.

Một vài kết quả của Outer Convex Approximation



Hình: Với $n = 100$, $\delta = 20$.

Một vài kết quả của Outer Convex Approximation



Hình: Với $n = 100$, $\delta = 50$.

Nội dung chính

1 Giới thiệu bài toán

- Đặt vấn đề
- Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)

2 Thuật toán tính bao lồi xấp xỉ

- Outer convex approximation

3 Thuật toán tính bao lồi trực giao

- Các khái niệm liên quan
- Thuật toán tìm bao lồi - Quickhull

4 Thực nghiệm và kết quả

- Môi trường

Nội dung chính

- 1 Giới thiệu bài toán
 - Đặt vấn đề
 - Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)
- 2 Thuật toán tính bao lồi xấp xỉ
 - Outer convex approximation
- 3 Thuật toán tính bao lồi trực giao
 - Các khái niệm liên quan
 - Thuật toán tìm bao lồi - Quickhull
- 4 Thực nghiệm và kết quả
 - Môi trường

Bao lỗi trực giao liên thông

Tính chất của bao lỗi trực giao liên thông

Đường trực giao bên phải

Nội dung chính

- 1 Giới thiệu bài toán
 - Đặt vấn đề
 - Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)
- 2 Thuật toán tính bao lồi xấp xỉ
 - Outer convex approximation
- 3 Thuật toán tính bao lồi trực giao
 - Các khái niệm liên quan
 - Thuật toán tìm bao lồi - Quickhull
- 4 Thực nghiệm và kết quả
 - Môi trường

Thuật toán Quickhull

Tính đúng đắn và phức tạp của Quickhull

Nội dung chính

- 1 Giới thiệu bài toán
 - Đặt vấn đề
 - Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)
- 2 Thuật toán tính bao lồi xấp xỉ
 - Outer convex approximation
- 3 Thuật toán tính bao lồi trực giao
 - Các khái niệm liên quan
 - Thuật toán tìm bao lồi - Quickhull
- 4 Thực nghiệm và kết quả
 - Môi trường

Nội dung chính

1 Giới thiệu bài toán

- Đặt vấn đề
- Phương pháp thích ứng bao lồi (convex-hull feature adaptation-CFA)

2 Thuật toán tính bao lồi xấp xỉ

- Outer convex approximation

3 Thuật toán tính bao lồi trực giao

- Các khái niệm liên quan
- Thuật toán tìm bao lồi - Quickhull

4 Thực nghiệm và kết quả

- Môi trường

Thay thế bằng thuật toán Outer Convex Approximation

Thay cả 2 hàm Jarvis() và Jarvis_and_index(), sử dụng full dataset