|  |  |
| --- | --- |
| **bỘ GIÁO DỤC VÀ ĐÀO TẠO** | **BỘ NÔNG NGHIỆP VÀ PTNT** |

TRƯỜNG ĐẠI HỌC THỦY LỢI



LÂM THỊ KIỀU

NÂNG CAO CHẤT LƯỢNG ẢNH TỐI DỰA TRÊN GIẢI THUẬT TỐI ƯU ĐỘNG VẬT BIỂN ĂN THỊT

ĐỒ ÁN TỐT NGHIỆP

HÀ NỘI, NĂM 2023

**LÂM THỊ KIỀU ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HÀ NỘI, NĂM 2023**

|  |  |
| --- | --- |
| **bỘ GIÁO DỤC VÀ ĐÀO TẠO** | **BỘ NÔNG NGHIỆP VÀ PTNT** |

TRƯỜNG ĐẠI HỌC THỦY LỢI

LÂM THỊ KIỀU

**NÂNG CAO CHẤT LƯỢNG ẢNH TỐI DỰA TRÊN GIẢI THUẬT TỐI ƯU ĐỘNG VẬT BIỂN ĂN THỊT**

|  |  |
| --- | --- |
| Ngành: | Hệ thống thông tin |
| Mã số: | 7480104 |

|  |  |
| --- | --- |
| NGƯỜI HƯỚNG DẪN | ThS. Đinh Phú Hùng |

HÀ NỘI, NĂM 2023

|  |  |
| --- | --- |
| Logo-WRU | CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM  **Độc lập - Tự do - Hạnh phúc**  ----------★----------  **NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP** |

**Họ tên sinh viên:**Lâm Thị Kiều **Hệ đào tạo**: Đại học chính quy

**Lớp**: 61HT **Ngành**: Hệ thống thông tin

**Khoa**: Công nghệ thông tin

1- TÊN ĐỀ TÀI:

NÂNG CAO CHẤT LƯỢNG ẢNH TỐI DỰA TRÊN GIẢI THUẬT TỐI ƯU ĐỘNG VẬT BIỂN ĂN THỊT

2- CÁC TÀI LIỆU

|  |  |
| --- | --- |
| [1] | A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, “Marine Predators Algorithm: A nature-inspired metaheuristic”, Expert Syst. Appl., vol. 152, 2020, doi: 10.1016/j.eswa.2020.113377. |
| [2] | Zuiderveld, K. (1994). Contrast limited adaptive histogram equalization. Graphics gems, 474-485. |
| [3] | C. Tomasi and R. Manduchi, " Bilateral filtering for gray and color images," Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271), 07-07 January 1998. |
| [4] | J. F. Canny, "A Computational Approach to Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vols. PAMI-8, no. 6, pp. 679 - 698, November 1986. |

3 - NỘI DUNG CÁC PHẦN THUYẾT MINH VÀ TÍNH TOÁN:

|  |  |
| --- | --- |
| Nội dung các phần | Tỷ lệ |
| Chương 1 - Tổng quan về tăng cường ảnh tối và kiến thức cơ bản | 30% |
| Chương 2 - Thuật toán tối ưu hóa động vật biển ăn thịt (Marine Predators Algorithm) | 35% |
| Chương 3 - Ứng dụng và thực nghiệm MPA để tăng cường ảnh tối | 35% |

4. GIÁO VIÊN HƯỚNG DẪN TỪNG PHẦN

|  |  |
| --- | --- |
| **Phần** | **Họ tên giáo viên hướng dẫn** |
| Chương 1 - Tổng quan về tăng cường ảnh tối và kiến thức cơ bản | Ths. Đinh Phú Hùng |
| Chương 2 - Thuật toán tối ưu hóa động vật biển ăn thịt (Marine Predators Algorithm) |
| Chương 3 - Ứng dụng và thực nghiệm MPA để tăng cường ảnh tối |

5. NGÀY GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Ngày 26 tháng 6 năm 2023

|  |  |
| --- | --- |
| **Trưởng Bộ môn**  *(Ký và ghi rõ Họ tên)* | **Giáo viên hướng dẫn chính**  *(Ký và ghi rõ Họ tên)* |

ThS. Đinh Phú Hùng

Nhiệm vụ Đồ án tốt nghiệp đã được Hội đồng thi tốt nghiệp của Khoa thông qua

Ngày. . . . .tháng. . . . .năm 2023 **Chủ tịch Hội đồng**

*(Ký và ghi rõ Họ tên)*

Sinh viên đã hoàn thành và nộp bản Đồ án tốt nghiệp cho Hội đồng thi ngày. 26 tháng 6 năm 2023.

**Sinh viên làm Đồ án tốt nghiệp**

*(Ký và ghi rõ Họ tên)*

Kiều

Lâm Thị Kiều

|  |  |
| --- | --- |
| logo | TRƯỜNG ĐẠI HỌC THUỶ LỢI  **KHOA CÔNG NGHỆ THÔNG TIN**  BẢN TÓM TẮT ĐỀ CƯƠNG ĐỒ ÁN TỐT NGHIỆP |

**TÊN ĐỀ TÀI:** **NÂNG CAO CHẤT LƯỢNG ẢNH TỐI DỰA TRÊN GIẢI THUẬT TỐI ƯU ĐỘNG VẬT BIỂN ĂN THỊT**

|  |  |
| --- | --- |
| *Sinh viên thực hiện:* | Lâm Thị Kiều |
| *Lớp:* | 61HT |
| *Giáo viên hướng dẫn:* | ThS. Đinh Phú Hùng |

Ảnh số đóng một phần quan trọng trong đời sống con người nó cung cấp thông tin về vật thể, hành động. Ảnh tối hay ảnh thiếu ánh sáng sẽ làm mất đi một phần thông tin đó vì vậy việc tăng cường chất lượng ảnh tối đóng một vai trò quan trọng trong nhiều ứng dụng của hình ảnh. Đồ án này đề xuất một phương pháp giúp tăng cường chất lượng ảnh tối dựa trên giải thuật tối ưu MPA (Marine Predators Algorithm) được đề xuất bởi A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi[[1]](#footnote-1).

Phương pháp đề xuất bao gồm 2 giai đoạn: đầu tiên, ảnh đầu vào sẽ được tăng cường độ tương phản thông qua các phương thức cơ bản: CLAHE[[2]](#footnote-2), bộ lọc Bilateral để khử nhiễu, Tạo ra ảnh được tăng cường về độ sắc nét bằng cách sử dụng toán tử Laplace, tạo ra 3 kênh tương ứng I1, I2, I3. Tiếp theo, sử dụng thuật toán MPA để tìm tham số phù hợp (α1 ,α2, α3) cho 3 kênh tương ứng I1, I2, I3. Cuối cùng thu được ảnh nâng cao chất lượng là tổng thành phần .

Xây dựng sản phẩm demo đầu vào là ảnh tối, thiếu ánh sáng, đầu ra ảnh được nâng cao chất lượng.

**CÁC MỤC TIÊU CHÍNH**

1. Tìm hiểu tổng quan một số phương pháp tăng cường ảnh tối, thiếu ánh sáng.
2. Tìm hiểu về thuật toán tối ưu động vật biển ăn thịt (MPA).
3. Tìm hiểu một số chỉ số đánh giá chất lượng ảnh.
4. Áp dụng thuật toán tối ưu hóa động vật biển ăn thịt vào phương pháp đề xuất để tăng cường chất lượng ảnh tối, thiếu ánh sáng
5. Xây dựng sản phẩm demo.

**KẾT QUẢ DỰ KIẾN**

Trình bày giải thuật đề xuất, xây dựng chương trình, đánh giá thực nghiệm và xây dựng sản phẩm.

**LỜI CAM ĐOAN**

Tác giả xin cam đoan đây là Đồ án tốt nghiệp của bản thân tác giả. Các kết quả trong Đồ án tốt nghiệp này là trung thực, và không sao chép từ bất kỳ một nguồn nào và dưới bất kỳ hình thức nào. Việc tham khảo các nguồn tài liệu (nếu có) đã được thực hiện trích dẫn và ghi nguồn tài liệu tham khảo đúng quy định.

|  |  |
| --- | --- |
|  | **Tác giả ĐATN**  *Chữ ký*  **Lâm Thị Kiều** |

**LỜI CẢM ƠN**

Trong suốt bốn năm học tập và nghiên cứu tại khoa Công nghệ thông tin, trường Đại học Thủy Lợi em đã nhận được rất nhiều sự quan tâm, hướng dẫn tận tình từ Thầy Cô và bạn bè, không chỉ tạo điều kiện cho chúng em phát triển và học tập bằng các môn học trên lớp mà còn cho chúng em những trải nhiệm nghiên cứu trong các cuộc thi như nghiên cứu khoa học, các bài tập lớn trên lớp, cuộc thi khởi nghiệp, …

Em xin gửi lời cảm ơn sâu sắc nhất đến thầy cô trong khoa nói chung và Thạc sĩ Đinh Phú Hùng nói riêng, người đã tận tâm chỉ bảo, giúp đỡ em hoàn Thành tốt đồ án tốt nghiệp này. Em xin gửi lòng biết ơn sâu sắc đến các thầy cô trong Khoa công nghệ thông tin đã chuyền đạt những kỹ năng và kiến thức quý báu ngay cả khi dịch Covid 19 làm đảo lộn nền giáo dục thầy cô không ngại khó không ngại khổ vẫn tận tâm truyền đạt kiến thức qua màn hình. Bốn năm trôi nhanh như một cơn gió, từ những ngày còn bỡ ngỡ đi nhập học đến giờ là môn học cuối cùng em cần phải vượt qua để có thể hoàn thành tốt bốn năm học tại trường. Những thầy cô đó những ký niệm đó em sẽ không bao giờ quên được.

Đồ án được em tận tâm thực hiện hết mình trong học phần này nhưng không thể tránh được một số thiếu sót, em rất mong sẽ nhận được sự đóng góp quý báu từ thầy cô và bạn bè để đồ án được hoàn thiện hơn nữa.

Em xin chân thành cảm ơn!

MỤC LỤC

[LỜI CAM ĐOAN i](#_Toc138654434)

[LỜI CẢM ƠN ii](#_Toc138654435)

[DANH MỤC CÁC HÌNH ẢNH v](#_Toc138654436)

[DANH MỤC CÁC BẢNG vi](#_Toc138654437)

[DANH MỤC CÁC TỪ VIẾT TẮT vii](#_Toc138654438)

[CHƯƠNG 1 TỔNG QUAN VỀ TĂNG CƯỜNG HÌNH ẢNH TỐI 3](#_Toc138654439)

[1.1 Tổng quan 3](#_Toc138654440)

[1.1.1 Đặt vấn đề 3](#_Toc138654441)

[1.1.2 Tổng quan về vấn đề nghiên cứu 4](#_Toc138654442)

[1.1.3 Lý do chọn đề tài 6](#_Toc138654443)

[1.2 Dữ liệu thực nghiệm 7](#_Toc138654444)

[1.3 Các chỉ số đánh giá chất lượng ảnh 9](#_Toc138654445)

[1.3.1 Độ sáng của ảnh. 9](#_Toc138654446)

[1.3.2 Độ tương phản của ảnh 9](#_Toc138654447)

[1.3.3 Nội dung thông tin chứa trong ảnh. 10](#_Toc138654448)

[1.3.4 Độ sắc nét của ảnh. 11](#_Toc138654449)

[CHƯƠNG 2 KIẾN THỨC NỀN TẢNG 12](#_Toc138654450)

[2.1 Biến đổi không gian màu 13](#_Toc138654451)

[2.1.1 Phép chuyển đổi từ hệ màu RGB sang hệ màu YUV 13](#_Toc138654452)

[2.1.2 Phép biến đổi từ hệ màu YUV sang hệ màu RGB. 15](#_Toc138654453)

[2.2 Một số phép tăng cường cơ bản 17](#_Toc138654454)

[2.2.1 Cân bằng histogram thích nghi (CLAHE) 17](#_Toc138654455)

[2.2.2 Lọc nhiễu bằng bộ lọc Bilateral 19](#_Toc138654456)

[2.2.3 Dò biên Canny 21](#_Toc138654457)

[2.3 Thuật toán tối ưu động vật biển ăn thịt MPA 23](#_Toc138654458)

[2.3.1 Ý tưởng chính 23](#_Toc138654459)

[2.3.2 Các bước giải thuật 24](#_Toc138654460)

[2.3.3 Sơ đồ thuật toán MPA 27](#_Toc138654461)

[2.3.4 Ví dụ minh họa 27](#_Toc138654462)

[CHƯƠNG 3 ĐỀ XUẤT PHƯƠNG PHÁP TĂNG CƯỜNG CHẤT LƯỢNG ẢNH 58](#_Toc138654463)

[3.1 Mô hình đề xuất 58](#_Toc138654464)

[3.2 Thực nghiệm và đánh giá 59](#_Toc138654465)

[3.2.1 Môi trường thực nghiệm 59](#_Toc138654466)

[3.2.2 Dữ liệu thực nghiệm 60](#_Toc138654467)

[3.3 Kết quả thực nghiệm 61](#_Toc138654468)

[KẾT LUẬN 68](#_Toc138654469)

[TÀI LIỆU THAM KHẢO 69](#_Toc138654470)

# DANH MỤC CÁC HÌNH ẢNH

[Hình 1.1 Dữ liệu thực nghiệm ảnh từ A1- A30 8](#_Toc138675472)

[Hình 2.1 Hình ảnh trước và sau khi cân bằng CLAHE 19](#_Toc138675473)

[Hình 2.2 Hình ảnh trước và sau khi sử dụng bộ lọc Bilateral 21](#_Toc138675474)

[Hình 2.3 Hình ảnh trước và sau khi thực hiện phép dò biên Canny 23](#_Toc138675475)

[Hình 2.4. Sơ đồ thuật toán MPA 27](#_Toc138675476)

[Hình 3.1 Các bước của kỹ thuật tăng cường ảnh 58](#_Toc138675477)

[Hình 3.2. Ảnh A1 trước và sau khi tăng cường 63](#_Toc138675478)

[Hình 3.3. Ảnh A2 trước và sau khi tăng cường 63](#_Toc138675479)

[Hình 3.4. Ảnh A3 trước và sau khi tăng cường 64](#_Toc138675480)

[Hình 3.5. Kết quả của các phương pháp tăng cường ảnh tối (ảnh minh hoạ A1-A5) 65](#_Toc138675481)

[Hình 3.6 Hình minh hoạ cho 4 chỉ số đánh giá của các phương pháp tăng cường ảnh khác nhau 66](#_Toc138675482)

# DANH MỤC CÁC BẢNG

[Bảng: 3‑1 Một số giải thuật tối ưu được sử dụng để so sánh 60](#_Toc138675446)

[Bảng: 3‑2 Một số giải thuật nâng cao chất lượng ảnh được so sánh 61](#_Toc138675447)

[Bảng: 3‑3 Giá trị trung bình và độ lệch chuẩn của các giải thuật so sánh 62](#_Toc138675448)

[Bảng: 3‑4 Giá trị trung bình các chỉ số của một số phương pháp tăng cường ảnh 66](#_Toc138675449)

[Bảng: 3‑5 Giá trị trung bình tham số tối ưu 67](#_Toc138675450)

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| 1 | YUV | Y là Luminance (Độ chói), là thuộc tính đặc trưng cho cường độ sáng, U và V là Chrominance (Sắc độ) |
| 2 | HE | Histogram equalization |
| 3 | RGB | Hệ màu Red – Green - Blue (Đỏ - Lục – Lam) |
| 4 | CLAHE | Contrast Limited Adaptive Histogram Equalization |
| 5 | MPA | Marine Predators Algorithm |
| 6 | DHE | Dynamic histogram equalization |
| 7 | CLDQHE | Contrast limited dynamic quadri-histogram equalization |
| 8 | ABHE | Daptive bi-histogram equaliza- tion |
| 9 | PSO | Particle swarm optimized |
| 10 | IFS | Intuitionistic fuzzy sets |
| 11 | CNN | Convolutional Neural Network |
| 12 | SR | Sparse Representation |
| 13 | ST | Shearlet Transform |
| 14 | FLA-Net | Fast and Lightweight Architecture Network |
| 15 | GAN | Generative Adversarial Network |
| 16 | AEMS | Attention Enhancement Network of Modules Stacking for Lowlight Image Enhancement |
| 17 | EOA | Equilibrium optimization algorithm |
| 18 | GOA | Grasshopper optimization algorithm |
| 19 | CSA | Chameleon swarm algorithm |
| 20 | PSO | Particle swarm optimization |
| 21 | SHO | Selfish herd optimizer |
| 22 | FPA | Flower pollination algorithm |
| 23 | SSA | Salp swarm algorithm |
| 24 | COA | Chimp optimization algorithm |
| 25 | BHE | Bi‐histogram equalization |

**MỞ ĐẦU**

Xử lý ảnh là một lĩnh vực quan trọng của khoa học máy tính, nó là quá trình xử lý hình ảnh đầu vào bằng các phương pháp tính toán và các thuật toán để tạo ra các thông tin từ ảnh ban đầu. Xử lý ảnh có ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau trong đời sống và công nghiệp hiện đại, như: xử lý ảnh trong y tế giúp hỗ trợ chuẩn đoán bệnh, nhận diện khuôn mặt, xử lý ảnh trong sản xuất kiểm tra chất lượng, xử lý ảnh trong thị giác máy tính, …Một lĩnh vực đặc biệt của xử lý ảnh là xử lý ảnh tối, nó tập trung vào xử lý những hình ảnh bị nhiễu, mờ, thiếu ánh sáng. Việc xử lý cải thiện ảnh tối còn đóng vai trò quan trọng trong đời sống con người khi mà xã hội con người ngày nay đã rất quen thuộc với mạng xã hội nơi mà 80% được chuyển tải bằng hình ảnh video. Chính vì vậy việc cải thiện chất lượng ảnh tối mang một ý nghĩa to lớn trong việc chuyển tải thông tin đầy đủ và chính xác hơn.

Một phương pháp mới được đề xuất trong đồ án này giúp cài thiện những hình ảnh được chụp trong điều kiện thiếu ánh sáng, ngoài ra phương pháp còn giải quyết được các vấn đề như nhiễu ảnh, hay hình ảnh bị làm sáng quá mức, đảm bảo được thông tin truyền tải của hình ảnh vẫn được giữ nguyên và đầy đủ. Ban đầu ảnh tối sẽ được tăng cường độ tương phản bẳng kỹ thuật CLAHE, kỹ thuật này giúp cải thiện độ tương phản của hình ảnh mà không làm mất đi các chi tiết quan trọng. Bộ lọc Bilateral khở nhiễu và giúp làm mịn ảnh trong khi vẫn giữ được các cạnh và chi tiết. Sau đó hình ảnh sẽ được xử lý bằng dò biên Canny để tăng cường độ nét. Sau cùng sử dụng thuật toán tối ưu Động vật biển ăn thịt (Marine Predators Algorithm) để tối ưu hóa các chỉ số từ ba phương pháp trên là α1, α2, α3. Để thu được hình ảnh đã tăng cường chất lượng từ các phương pháp CLAHE, Bilateral và Canny (I1, I2, I3) cần tìm được tham số α1, α2, α3 phù hợp nhất. Ảnh cuối cùng thu được là tổng các thành phần

Dựa trên thuật toán tối ưu động vật biển ăn thịt (MPA) do A. Faramarzi và các đồng nghiệp [1] đề xuất, trong đồ án này trình bày các bước giải thuật mà tác giả đề xuất và xây dựng chương trình minh họa.

Bố cục đồ án gồm 3 chương:

**Chương 1:** Tổng quan về tăng cường hình ảnh tối

**Chương 2 :** Kiến thức cơ bản và giải thuật MPA (Marine Predators Algorithm)

**Chương 3**: Kết quả thực nghiệm

# TỔNG QUAN VỀ TĂNG CƯỜNG HÌNH ẢNH TỐI

## Tổng quan

### Đặt vấn đề

Xử lý ảnh là một lĩnh vực quan trọng của khoa học máy tính và phát triển rất nhanh trong thời gian gần đây. Những tiến bộ trong công nghệ đã mở ra những cánh cửa mới cho xử lý ảnh đưa ra nhiều cơ hội và thách thức để áp dụng những kỹ thuật mới vào các ứng dụng khác nhau nhưu xử lý ảnh kỹ thuật số, phát hiện đối tượng và nhận dạng, xử lý ảnh thời gian thực, học sâu và trí tuệ nhân tạo. Nâng cao chất lượng ảnh tối là một trong những chủ đề quan trọng trong thị giác máy tính, được áp dụng rộng rãi trong việc nâng cao chất lượng ảnh màu, trong lĩnh vực y tế: việc tăng cường chất lượng ảnh có thể cải thiện khả năng chuẩn đoán theo dõi các bệnh lý của bệnh nhân. Trong lĩnh vực an ninh giúp tìm ra điểm đặc biệt, chi tiết quan trọng, giám sát được thay đổi của đối tượng. Trong lĩnh vực nhiếp ảnh việc nâng cao chất lượng ảnh có thể giúp nhiếp ảnh gia tạo ra những bức ảnh đẹp hơn, chuyên nghiệp hơn, … và nhiều lĩnh vực khác.

Ảnh tối hoặc thiếu ánh sáng thường được chụp trong những điều kiện ánh sáng kém hoặc trong môi trường thiếu ánh sáng. Đặc điểm ảnh là có độ sáng thấp, độ tương phản thấp, độ nhiễu cao, khó nhận diện được các chi tiết và thông tin trong ảnh. Ảnh tối có thể gây ra nhiều nhược điểm trong an ninh, chẳng hạn như: Làm giảm hiệu quả của các hệ thống nhận diện khuôn mặt, biển số xe, mã vạch, … Điều này có thể dẫn đến sai sót, lỗi và rủi ro trong việc kiểm soát an ninh, quản lý hàng hóa và dịch vụ. Ví dụ, ảnh tối có thể làm cho hệ thống nhận diện khuôn mặt không thể phân biệt được các đối tượng khác nhau, hoặc nhầm lẫn giữa các đối tượng có khuôn mặt tương tự; Trong an ninh y tế, ảnh tối có thể làm giảm chất lượng của các hình ảnh y khoa, ảnh chụp X-quang, siêu âm, MRI, …. Điều này có thể ảnh hưởng đến việc chuẩn đoán bệnh, theo dõi sức khỏe và điều trị của bệnh nhân. Ngoài ra, ảnh tối cũng có thể làm khó việc sử dụng trí tuệ nhân tạo để phân tích và xử lý ảnh y khoa.

Việc chụp ảnh trong điều kiện thiếu ánh sáng không chỉ làm cho các chi tiết trên ảnh bị mất dần, nên khó nhìn thấy mà đồng thời màu sắc trên ảnh có thể bị biến đổi hoặc mất đi do sự khác biệt ánh sáng giữa các khu vực trên bức ảnh. Ngoài ra, các chi tiết màu trên ảnh có thể bị ảnh hưởng bởi độ phân giải kém hoặc nhiễu hạt. Những bức ảnh tối và thiếu ánh sáng sẽ gây khó khăn trong việc sử dụng và xử lý, đặc biệt trong các hoạt động như chỉnh sửa ảnh hoặc phục hồi ảnh cũ, ngoài ra thông tin sai lệch từ hình ảnh có thể ảnh hưởng nghiêm trọng đến việc tiếp nhận thông tin liên quan. Khi sử dụng các bức ảnh này, thông tin trên ảnh có thể bị mất mát hoặc bị sai lệch, gây ra các hiểu nhầm hoặc thông tin không chính xác. Việc khắc phục khá khó khăn và mất nhiều thời gian. Việc tăng cường ảnh tối sẽ tạo ra được hình ảnh có yếu tố như độ tương phản, độ sắc nét, nhiễu sẽ cải thiện hơn, tốt hơn.

### Tổng quan về vấn đề nghiên cứu

Để giải nâng cao chất lượng của hình ảnh tối cần giải quyết ba vấn đề: Tăng cường độ tương phản, giảm nhiễu, làm sắc nét hình ảnh. Có hai nhóm chính các phương pháp tiếp cận để nâng cao hình ảnh: các phương pháp dựa trên miền không gian và các phương pháp dựa trên miền chuyển đổi. Các kỹ thuật dựa trên miền không gian thao tác trực tiếp trên pixel và khối pixel. Các phương pháp này có ưu điểm là đơn giản và có độ phức tạp tính toán thấp. Cân bằng biểu đồ (HE) là một phương pháp truyền thống để cải thiện độ sáng và độ tương phản của hình ảnh. Tuy nhiên, phương pháp HE có nhược điểm là làm tăng cường hình ảnh quá mức và tạo ra nhiễu. Do đó, nhiều thuật toán khác đã được đề xuất để khắc phục hạn chế của phương pháp HE, như CLAHE [2], DHE [3] và CLDQHE [4]. Ngoài ra, nhiều nghiên cứu mới về tăng cường hình ảnh cũng dựa trên sự cải tiến của phương pháp HE. Ví dụ, Paul và đồng nghiệp đã đề xuất ABHE [5] để cải thiện độ tương phản của hình ảnh. Mishiro và cộng sự [6] đã giới thiệu một phương pháp mới dựa trên HE để cải thiện độ tương phản của MRI não. Acharya và cộng sự [7] đã kết hợp PSO với HE để nâng cao hình ảnh não MRI. Jebadass và cộng sự [8] kết hợp IFS với HE. Vijayalak-shmi và cộng sự [9] đã đề xuất một thuật toán mới để tăng cường độ tương phản bằng cách kết hợp độ dốc với HE.

Để nâng cao hình ảnh dựa trên miền biến đổi, thường có ba bước được thực hiện. Bước đầu tiên là chuyển ảnh từ miền không gian sang miền biến đổi bằng một thuật toán biến đổi ảnh. Bước thứ hai là tăng cường các thành phần trên miền biến đổi bằng các thuật toán cụ thể. Bước cuối cùng là chuyển các thành phần đã được tăng cường trở lại miền không gian để thu được hình ảnh đã được nâng cao. Nhiều cách tiếp cận dựa trên miền chuyển đổi đã được đề xuất cho đến nay. Ví dụ, Huang và cộng sự [10] đã sử dụng miền biến đổi đường viền để nâng cao chất lượng hình ảnh. Goyal và cộng sự [11] đã khử nhiễu hình ảnh trên miền biến đổi shearlet không được lấy mẫu phụ. Kollem và cộng sự [12] đã khử nhiễu hình ảnh y tế trên miền biến đổi đường biên không được lấy mẫu phụ. Sahnoun và cộng sự [13] hình ảnh MRI nâng cao trong miền biến đổi sóng con rời rạc. Ngoài ra, biểu diễn thưa cũng là một công cụ hữu hiệu để nâng cao chất lượng ảnh. Lei và cộng sự [14] đã khử nhiễu hình ảnh tĩnh mạch ngón tay bằng biểu diễn thưa thớt (SR). Zhou và cộng sự [15] đã kết hợp tính tự tương tự không định hướng nâng cao với SR để khử nhiễu hình ảnh mới.

Học sâu cũng là một phương pháp mới được nhiều nghiên cứu mới đề xuất để nâng cao hình ảnh. Ví dụ, Lyu và cộng sự [16] đã khử nhiễu hình ảnh bằng mô hình mạng nơ ron tích chập (CNN) trên miền biến đổi shearlet (ST). Yan và cộng sự [17] đã kết hợp Non-local self-similarity với CNN để khử nhiễu hình ảnh. Yu và cộng sự [18] đã giới thiệu một mạng học sâu mới tên là FLA-Net để cải thiện hình ảnh thiếu sáng. Wang và cộng sự [19] đã sử dụng Mạng đối thủ sáng tạo (GAN) để nâng cao chất lượng hình ảnh thiếu sáng. Li và cộng sự [20] đã đề xuất một mạng tên là AEMS để cải thiện hình ảnh thiếu sáng.

Thuật toán tối ưu hóa là công cụ mạnh nhất hiện nay để giải quyết các bài toán xử lý ảnh. Các thuật toán tối ưu hóa đã được áp dụng cho vấn đề kết hợp các hình ảnh y tế, ví dụ như MPA [21] [22], EOA [23], [24], GOA [25] và CSA [26]. Các thuật toán tối ưu hóa cũng đã được sử dụng để nâng cao chất lượng hình ảnh. Ví dụ, Sathananthavathi và cộng sự [27] đã sử dụng PSO để nâng cao chất lượng hình ảnh võng mạc. Wadhwa và cộng sự [28] đã kết hợp các biến đổi hình thái và PSO để tăng cường độ tương phản của hình ảnh MRI. Zhou và cộng sự. Guha và cộng sự [29] đã áp dụng SHO để cải thiện chất lượng hình ảnh. Ngoài ra, còn có nhiều thuật toán tối ưu hóa khác như FPA [30], SSA [31], COA [32].

Giải thuật được Faramarzi và cộng sự [1] đề xuất. Thuật toán MPA đã được sử dụng trong nhiều ứng dụng xử lý hình ảnh như hợp nhất hình ảnh y tế [22], phân đoạn hình ảnh [33] và tăng cường hình ảnh [34], và đã cho thấy hiệu quả. Do đó, trong nghiên cứu này, mô hình đề xuất sử dụng thuật toán MPA. Tuy nhiên, một hình ảnh có thể bị ảnh hưởng bởi nhiễu, độ tương phản kém và độ sắc nét không cao cùng một lúc. Các phương pháp tăng cường hiện nay chỉ tập chung vào giải quyết các vẫn đề riêng lẻ như độ tương phản hay giảm nhiễu ảnh tối hoặc tăng cường độ sắc nét hình ảnh. Ngoài ra, tiền xử lý ảnh là một bước quan trọng trong nhiều bài toán xử lý ảnh. Ví dụ, Ullah và cộng sự [35] ​​đã cải thiện chất lượng của hình ảnh đầu vào để phân loại hình ảnh MRI não. Veluchamy và cộng sự [36] sử dụng BHE để tiền xử lý hình ảnh trước khi phân đoạn.

Nâng cao chất lượng ảnh tối là một yếu tố quan trọng trong lĩnh vực xử lý ảnh, với tầm quan trọng không chỉ giới hạn trong việc cải thiện hình ảnh, mà còn ảnh hưởng sâu rộng đến nhiều khía cạnh trong cuộc sống.Việc nâng cao chất lượng ảnh tối giúp tăng khả năng nhìn thấy và phân tích thông tin. Trong môi trường ánh sáng yếu, ảnh tối thường gây mất mát chi tiết quan trọng. Nâng cao chất lượng ảnh tối cũng góp phần quan trọng vào trải nghiệm người dùng. Trong lĩnh vực chụp ảnh và quay video, chất lượng ảnh tối cao đảm bảo rằng người dùng có thể ghi lại những khoảnh khắc đáng nhớ trong môi trường ánh sáng yếu mà vẫn thu được hình ảnh sắc nét và rõ ràng. Điều này tạo ra sự hài lòng và thỏa mãn cho người dùng, đồng thời nâng cao trải nghiệm của họ khi sử dụng các thiết bị và ứng dụng liên quan đến hình ảnh. Trong lĩnh vực an ninh và giám sát. Các hệ thống giám sát an ninh yêu cầu khả năng nhìn rõ và nhận dạng đối tượng, sự kiện trong môi trường ánh sáng yếu. Nhờ việc cải thiện chất lượng ảnh tối, các hệ thống này có thể theo dõi và xác định các hoạt động gian lận, trộm cắp hoặc các hành vi đáng ngờ, đảm bảo an ninh và an toàn cho cộng đồng và các tổ chức. Hình ảnh chất lượng cao trong điều kiện ánh sáng yếu giúp con người chia sẻ những khoảnh khắc quan trọng, truyền tải thông điệp một cách rõ ràng và chân thực hơn. Điều này có ý nghĩa quan trọng trong việc truyền tải thông tin, truyền đạt cảm xúc và kết nối với người khác thông qua hình ảnh.

### Lý do chọn đề tài

Ảnh tối được chụp trong điều kiện ánh sáng kém, buổi tối, hay thời tiết âm u, ngược sáng, ảnh tối thường bị suy giảm về nội dung thông tin, màu sắc, nhiễu hạt và độ sắc nét kém. Việc nâng cao chất lượng ảnh tối không chỉ cải thiện hình ảnh mà còn có tầm quan trọng to lớn trong nhiều khía cạnh của cuộc sống và công việc. Từ việc cải thiện khả năng nhìn thấy và phân tích thông tin, tăng trải nghiệm người dùng, đến cải thiện an ninh và giám sát, cũng như tăng cường khả năng diễn đạt và truyền tải thông tin, nâng cao chất lượng ảnh tối mang lại nhiều lợi ích và ứng dụng thực tế. Để phân tích và trích xuất thông tin từ một bức ảnh tối đạt hiệu quả cao thì một yêu cầu bức thiết là nâng cao chất lượng hay tăng cường nội dung thông tin chứa trong bức ảnh đó. Tăng cường chất lượng ảnh có thể coi là khâu tiền xử lý cho các bước tiếp theo trong quy trình xử lý ảnh. Nâng cao chất lượng ảnh tối không còn là một bài toán mới, có rất nhiều phương pháp để giải quyết bài toán này. Từ việc nghiên cứu các phương pháp đã có như các phương pháp truyền thống đến nâng cao, gần đây nhất có kết hợp với các thuật toán tăng cường nhưng mỗi phương pháp đều có nhược điểm là chỉ tập chung và giải quyết một vấn đề riêng lẻ. Với những hạn chế đó, đồ án này sẽ đề xuất một phương pháp mới tăng cường ảnh tối có thể cải thiện đồng thời ba vấn đề về ánh áng, nhiễu , sắc nét của ảnh tối, trong đó sử dụng giải thuật tối ưu hoá động vật biển ăn thịt MPA (Marine Predators Algorithm) được đề xuất bởi A. Faramarzi và các đồng nghiệp [1] để tìm ra các tham số tối ưu cho phương pháp đề xuất. Phương pháp này sẽ giúp hạn chế việc làm giảm chất lượng ảnh, giảm cường độ sáng, giảm độ mờ của ảnh và đánh mất thông tin của ảnh.

## Dữ liệu thực nghiệm

Dữ liệu thử nghiệm được lấy từ Bộ dữ liệu nguồn LOL[[3]](#footnote-3). Nguồn dữ liệu được sử dụng trong thư mục **low** của thử mục **our485** gồm 485 ảnh tối. những bức ảnh được chụp từ camera với độ sáng khác nhau, hình ảnh chủ yếu là các vật thể, quang cảnh… Trong nghiên cứu này sẽ sử dụng 30 ảnh để thực hiện thực nhiệm.

Hình 1.1 là 30 ảnh trong bộ dữ liệu sẽ được sử dụng trong nghiên cứu này. Hình ảnh sẽ được đặt theo tên: A1 đến A30:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| A1 | A2 | A3 | A4 | A5 |
|  |  |  |  |  |
| A6 | A7 | A8 | A9 | A10 |
|  |  |  |  |  |
| A11 | A12 | A13 | A14 | A15 |
|  |  |  |  |  |
| A16 | A17 | A18 | A19 | A20 |
|  |  |  |  |  |
| A21 | A22 | A23 | A24 | A25 |
|  |  |  |  |  |
| A26 | A27 | A28 | A29 | A30 |

Hình 1.1 Dữ liệu thực nghiệm ảnh từ A1- A30

## Các chỉ số đánh giá chất lượng ảnh

### Độ sáng của ảnh.

Giá trị trung bình của ảnh là chỉ số về độ sáng của ảnh.

Công thức tính giá trị trung bình của ảnh là:

|  |  |
| --- | --- |
|  | (1‑1) |

Trong đó:

* là giá trị trung bình
* lần lượt tổng số hàng, cột của ma trận.
* là phần hàng u, cột v của ma trận.

Ví dụ: Ma trận A có dạng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | 20 | 10 | 40 |
| A = | 10 | 35 | 30 |
|  | 20 | 10 | 50 |

Ta thực hiện chuyển ma trận về miền [0, 1]:

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0.4 | 0.2 | 0.8 |
| A = | 0.2 | 0.7 | 0.6 |
|  | 0.4 | 0.2 | 1 |

Giá trị trung bình μ là:

### Độ tương phản của ảnh

Công thức tính độ tương phản (phương sai) của ảnh:

|  |  |
| --- | --- |
|  | (1‑2) |

Trong đó:

* là độ tương phản của ảnh.
* lần lượt tổng số hàng, cột của ma trận.
* là phần tử hàng u, cột v của ma trận

Ví dụ: Ma trận A có dạng

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0.4 | 0.2 | 0.8 |
| A = | 0.2 | 0.7 | 0.6 |
|  | 0.4 | 0.2 | 1 |

Giá trị phương sai là:

### Nội dung thông tin chứa trong ảnh.

Nội dung thông tin diễn tả lượng thông điệp mà bức ảnh chứa trong đó. Thông số này còn được gọi là entropy và được tính bằng công thức:

|  |  |
| --- | --- |
|  | (1‑3) |

Ví dụ: Ma trận A có dạng

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0.4 | 0.2 | 0.8 |
| A = | 0.2 | 0.7 | 0.6 |
|  | 0.4 | 0.2 | 1 |

Xây dựng bảng thống kê số lần xuất hiện của từng điểm ảnh:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Mức xám | 0.2 | 0 | 0.1 | 1 | 0.7 | 0.4 | 0.8 |
| Số lần xuất hiện |  |  |  |  |  |  |  |
| Xác suất |  |  |  |  |  |  |  |

Suy ra:

### Độ sắc nét của ảnh.

Độ sắc nét của ảnh được tính bằng công thức:

|  |  |
| --- | --- |
| và | (1‑4) |

Trong đó:

* và
* là giá trị điểm ảnh tại hang u, cột v của ảnh.
* là giá trị điểm ảnh tại hang , cột của ảnh
* là giá trị điểm ảnh tại hang , cột của ảnh
* M, N lần lượt tổng số hàng, cột của ma trận.

Ví dụ ma trận A có dạng:

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0.4 | 0.2 | 0.8 |
| A = | 0.2 | 0.7 | 0.6 |
|  | 0.4 | 0.2 | 1 |

|  |  |
| --- | --- |
|  |  |

Ta thu được:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0.2 | − 0.5 | 0.2 |  | 0.2 | − 0.6 | 0 |
| u = | − 0.2 | 0.5 | − 0.4 | v = | −0.5 | 0.1 | 0 |
|  | 0 | 0 | 0 |  | 0.2 | −0.8 | 0 |

Suy ra:

# KIẾN THỨC NỀN TẢNG

## Biến đổi không gian màu

### Phép chuyển đổi từ hệ màu RGB sang hệ màu YUV

Chuyển đổi từ RGB sang YUV trong đó hệ màu RGB (Red, Green, Blue) là một hệ màu được sử dụng rộng rãi trong hình ảnh số và các thiết bị hiển thị. Trong hệ màu RGB, mỗi pixel trong một hình ảnh được biểu diễn bằng ba thành phần màu: đỏ (Red), xanh lá cây (Green) và xanh dương (Blue). Hệ màu YUV là một hệ màu được sử dụng trong xử lý hình ảnh và video. Trong hệ màu này, thành phần Y đại diện cho độ sáng của một pixel, trong khi các thành phần U và V mang thông tin về màu sắc. YUV được sử dụng để phân tách thông tin màu sắc và độ sáng, giúp hiển thị và xử lý hình ảnh một cách hiệu quả. Thông tin độ sáng (Y) là quan trọng trong quá trình hiển thị, trong khi các thành phần màu (U và V) mang thông tin về màu sắc. Hệ màu YUV được áp dụng trong các chuẩn video và công nghệ hiển thị màu, và cung cấp cách tiếp cận linh hoạt cho xử lý hình ảnh và video.

Công thức chuyển đổi từ RGB sang YUV:

|  |  |
| --- | --- |
|  | (2‑1) |
|  | (2‑2) |
|  | (2‑3) |

Trong đó:

* Y là thành phần độ sáng,
* U là thành phần màu xanh dương
* V là thành phần màu đỏ.

**Ví dụ minh họa trên ma trận:**

Giả sử ảnh đầu vào trong hệ màu RGB có cỡ có ma trận giá trị các kênh màu lần lượt là:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |
| R = |  |  |  | G = |  |  |  | B = |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

Áp dụng công thức (2-1) ta thu được ma trận giá trị kênh Y là:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| Y = |  |  |  |
|  |  |  |  |

Trên thực tế ma trận ảnh là các số nguyên nên ta tiến hành làm tròn các số nguyên:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| Y = |  |  |  |
|  |  |  |  |

Xác định giá trị kênh màu U theo công thức (2‑2):

Ta thu được ma trận giá trị kênh U là:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| U = |  |  |  |
|  |  |  |  |

Xác định giá trị kênh màu V theo công thức (2-3):

Ta thu được ma trận giá trị kênh V là:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| V = |  |  |  |
|  |  |  |  |

### Phép biến đổi từ hệ màu YUV sang hệ màu RGB.

Chuyển đổi từ YUV sang RGB:

|  |  |
| --- | --- |
|  | (2‑4) |
|  | (2‑5) |
|  | (2‑6) |

Trong công thức trên, Y, U và V là các thành phần của không gian màu YUV.

**Ví dụ minh họa:**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| Y = |  |  |  |
|  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
| U = |  |  |  | V = |  |  |  |
|  |  |  |  |  |  |  |  |

Áp dụng công thức (2-4) ta thu được ma trận giá trị kênh R là:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| R = |  |  |  |
|  |  |  |  |

Trên thực tế ma trận ảnh là các số nguyên nên ta tiến hành làm tròn các số nguyên:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| R = |  |  |  |
|  |  |  |  |

Xác định giá trị kênh màu G theo công thức (2-5):

Ta thu được ma trận giá trị kênh G là:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| G = |  |  |  |
|  |  |  |  |

Trên thực tế ma trận ảnh là các số nguyên nên ta tiến hành làm tròn các số nguyên:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| G = |  |  |  |
|  |  |  |  |

Xác định giá trị kênh màu B theo công thức (2-6):

Ta thu được ma trận giá trị kênh B là:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| B = |  |  |  |
|  |  |  |  |

Trên thực tế ma trận ảnh là các số nguyên nên ta tiến hành làm tròn các số nguyên:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| B = |  |  |  |
|  |  |  |  |

## Một số phép tăng cường cơ bản

### Cân bằng histogram thích nghi (CLAHE)

Histogram của một ảnh với mức xám trong khoảng là một hàm rời rạc

|  |  |
| --- | --- |
|  | (2‑7) |

Trong đó:là mức xám thứ và là số lượng pixel trong ảnh có mức xám .

Histogram là một đồ thị biểu diễn độ sáng của một bức ảnh với trục hoành là độ sáng và trục tung là số lượng điểm ảnh ở độ sáng tương ứng. Chiều cao của các cột trên đồ thị cũng thể hiện số lượng pixel ở mức sáng tương ứng.

Histogram thường được chuẩn hóa. Với là tổng số pixels của ảnh, histogram chuẩn hóa được tính qua biểu thức:

|  |  |
| --- | --- |
|  | (2‑8) |

Tuy nhiên, Histogram có xu hướng khuếch đại quá mức nhiễu ở các vùng tương đối đồng nhất của hình ảnh. Một biến thể của cân bằng biểu đồ thích ứng được gọi là cân bằng biểu đồ thích ứng giới hạn tương phản (CLAHE) [2] ngăn chặn điều này bằng cách hạn chế khuếch đại.

Trong CLAHE, độ khuếch đại tương phản trong vùng lân cận của một giá trị pixel nhất định được cho bởi độ dốc của hàm biến đổi. Điều này tỷ lệ thuận với độ dốc của hàm phân phối tích lũy lân cận (CDF) và do đó với giá trị của biểu đồ tại giá trị pixel đó. CLAHE hạn chế khuếch đại bằng cách cắt biểu đồ ở một giá trị được xác định trước trước khi tính toán CDF. Điều này giới hạn độ dốc của CDF và do đó giới hạn hàm chuyển đổi. Giá trị tại đó biểu đồ được cắt bớt, cái gọi là giới hạn clip, phụ thuộc vào việc chuẩn hóa biểu đồ và do đó phụ thuộc vào kích thước của vùng lân cận.

Giải thuật:

Giả sử: Ảnh đầu vào có ma trận cỡ m × n và ma trận cửa sổ lọc cỡ 3 × 3.

**Bước 1:** Tính toán histogram

**Bước 2:** Tính toán đồng đều (cumulative histogram): Từ histogram, ta tính toán đồng đều bằng cách tích lũy tần suất xuất hiện từ mức xám đầu tiên cho đến mức xám đang xét. Đồng đều cung cấp thông tin về phân phối mức xám trong cửa sổ liều.

**Bước 3:** Tính toán giới hạn (limiting):

Giới hạn (L) được tính bằng công thức:

|  |  |
| --- | --- |
|  | (2‑9) |

Trong đó:

* clipLimit là một hệ số giới hạn (thường được đặt là một giá trị dương nhỏ, ví dụ: 2.0).
* S là tổng số pixel trong cửa sổ liều.
* N là số bin trong histogram.

**Bước 4:** Áp dụng trọng số và tính toán giá trị đầu ra.

Hình 2.1 Minh hoạ hình ảnh đầu vào là ảnh tối, và hình ảnh đầu ra sau khi sửa lý qua phép tăng cường ảnh CLAHE, Độ sáng, độ tương phản của ảnh được cải thiện đáng kể, chung ta có thể nhìn thấy sự thay đổi rõ rệt.

|  |  |
| --- | --- |
|  |  |
| Hình ảnh đầu vào | Hình ảnh cân bằng CLAHE |

Hình 2.1 Hình ảnh trước và sau khi cân bằng CLAHE

### Lọc nhiễu bằng bộ lọc Bilateral

Bộ lọc Bilateral hay còn gọi là bộ lọc hai chiều đây là một phương pháp để khử nhiễu trong xử lý ảnh được đề xuất bởi C. Tomasi; R. Manduchi vào năm 1998 [37] đạt hiệu quả cao được sử dụng để làm mịn ảnh trong khi vẫn giữ được các cạnh và chi tiết.

Ý tưởng chính của bộ lọc Bilateral là thay đổi giá trị độ sáng của các điểm ảnh trong ảnh bằng cách lấy trung bình có trọng số của các điểm ảnh xung quanh nó. Các điểm ảnh càng gần điểm ảnh đang xét thì trọng số càng lớn. Điều này có nghĩa là các điểm ảnh gần nhau và có giá trị màu tương tự sẽ có trọng số lớn hơn, trong khi các điểm ảnh xa và có giá trị màu khác biệt sẽ có trọng số nhỏ hơn. Việc sử dụng bộ lọc Bilateral có thể giúp loại bỏ các đốm nhiễu và giảm độ nhiễu trong ảnh, đồng thời giữ lại các cạnh và chi tiết quan trọng trong ảnh. Như chúng ta đã biết, bộ lọc Gauss quyết định giá trị một điểm ảnh bằng cách lấy trung bình theo hàm Gauss các giá trị điểm ảnh xung quanh điểm đó. Hàm trọng số Gauss chỉ phụ thuộc vào khoảng cách trong không gian so với điểm ảnh trung tâm, không quan tâm đến sự tương quan giữa mức xám của điểm trung tâm với các điểm xung quanh đó. Nó cũng không quan tâm rằng điểm ảnh trung tâm có nằm tại một đường biên trong ảnh không, vì thế làm nhòe luôn các đường biên trong ảnh.

Bộ lọc Bilateral cũng sử dụng một bộ lọc Gauss với khoảng cách đến điểm trung tâm, đảm bảo chỉ có các điểm ở gần tham gia vào giá trị của điểm ảnh trung tâm. Tuy vậy nó sử dụng thêm một hàm Gauss cho mức xám, đảm bảo chỉ các điểm ảnh có mức xám tương đồng với điểm ảnh trung tâm tham gia vào quá trình làm mịn. Vì thế bộ lọc Bilateral bảo toàn được các đường biên trong ảnh bởi vì điểm ảnh ở biên có sự thay đổi về mức xám rất rõ ràng. Bộ lọc Bilateral có thể được thực hiện bằng cách sử dụng một ma trận trọng số (kernel) được tính toán dựa trên độ tương đồng giữa các điểm ảnh. Ma trận trọng số này sẽ được áp dụng lên từng điểm ảnh trong ảnh, giúp tính toán giá trị trung bình mới cho điểm ảnh đó.

Các bước giải thuật:

Giả sử: Ảnh đầu vào có ma trận cỡ m × n và ma trận cửa sổ lọc cỡ 3 × 3.

**Bước 1:** Xác định vùng lân cận của điểm ảnh chính là ma trận 3x3 với trung điểm là điểm ảnh đầu vào

**Bước 2:** Tính toán trọng số cho mỗi điểm ảnh trong vùng lân cận:

|  |  |
| --- | --- |
|  | (2‑10) |

Trong đó:

* điểm ảnh lân cận
* là khoảng cách không gian giữa 2 điểm ảnh đó
* là khoảng cách độ sáng giữa 2 điểm ảnh đó

**Bước 3:** Tính tổng trọng số của tất cả các điểm lân cận

|  |  |
| --- | --- |
|  | (2‑11) |

**Bước 4:** Tính giá trị kết qủa đầu ra

|  |  |
| --- | --- |
|  | (2‑12) |

Trong đó:

* Là giá trị điểm ảnh tại vị trí p, q trên ảnh đầu vào

**Bước 5:** Áp dụng trọng số và tính toán giá trị đầu ra

Hình 2.2 Minh hoạ cho ảnh đầu vào có nhiễn và hình ảnh đầu ra sau khi sửa lý qua bộ lọc Bilateral, nhiễu của ảnh đã được cải thiện đáng kể thông qua bộ lọc cho thấy hiệu quả tốt của bộ lọc Bilateral.

|  |  |
| --- | --- |
|  |  |
| Hình ảnh 𝐼𝑖𝑛 đầu vào | Hình ảnh lọc bilateral |

Hình 2.2 Hình ảnh trước và sau khi sử dụng bộ lọc Bilateral

### Dò biên Canny

Canny là một phương pháp xử lý ảnh thường được sử dụng để tìm và trích xuất các biên của đối tượng trong ảnh. Phương pháp này được phát triển bởi John F. Canny vào năm 1986 [38] và được xem là một trong những phương pháp tiên tiến nhất trong lĩnh vực này. Nó được sử dụng rộng rãi trong nhiều ứng dụng xử lý ảnh, bao gồm nhận dạng khuôn mặt, nhận dạng vật thể, xử lý ảnh y học, tăng cường ảnh, và phát hiện vật thể trong thời gian thực. Phương pháp này cho phép trích xuất các đường viền và biên của vật thể trong ảnh, giúp cho việc phân tích và xử lý ảnh trở nên dễ dàng hơn.

Các bước của thuật toán:

**Bước 1**: Làm trơn ảnh.

Vì tất cả các kết quả phát hiện cạnh dễ bị ảnh hưởng bởi nhiễu hình ảnh, nên việc lọc nhiễu để tránh phát hiện sai do nhiễu. Để làm mịn hình ảnh, một hạt nhân bộ lọc Gaussian được kết hợp với hình ảnh. Bước này sẽ làm mịn hình ảnh một chút để giảm ảnh hưởng của nhiễu rõ ràng lên máy dò cạnh. Phương trình cho hạt nhân bộ lọc Gaussian có kích thước (2 k +1) × (2 k +1) được đưa ra bởi:

|  |  |
| --- | --- |
|  | (2‑13) |

Tính tích chập G = I  H, với

**Bước 2**: Tính Gradient của ảnh bằng mặt nạ Prewitt theo hai hướng x, y. Gọi là GX, GY.

**Bước 3**: Tính Gradient theo 8 hướng tương ứng với 8 lân cận của 1 điểm ảnh

**Bước 4**: Loại bỏ những điểm không phải cực nhằm xóa bỏ những điểm không thuộc biên

**Bước 5**: Phân ngưỡng. Thực hiện lấy Gradient lần cuối

Gọi là ảnh sau dò biên Canny giúp tìm các đường biên yếu của ảnh sử dụng hai mức ngưỡng cao và thấp. Ban đầu ta dùng mức ngưỡng cao để tìm điểm bắt đầu của biên, sau đó chúng ta xác định hướng phát triển của biên dựa vào các điểm ảnh liên tiếp có giá trị lớn hơn mức ngưỡng thấp. Ta chỉ loại bỏ các điểm có giá trị nhỏ hơn mức ngưỡng thấp. Các đường biên yếu sẽ được chọn nếu chúng được liên kết với các đường biên khỏe. Trong matlab, ta có thể sử dụng hàm để làm việc với dò biên Canny.

|  |  |
| --- | --- |
|  | (2‑14) |

Trong đó:

* f là ảnh ban đầu, có miền giá trị thuộc [0, 1].
* Canny là tham số của phương pháp dò biên Canny.

Hình 2.3 Minh hoạ với là ảnh đầu vào, ta thu được hình ảnh sau khi sửa lý qua dò biên Canny, vùng biên của ảnh được trích xuất riêng biệt thông qua dò biên Canny.

|  |  |
| --- | --- |
|  |  |
| Hình ảnh 𝐼𝑖𝑛 đầu vào | Hình ảnh dò biên Canny 𝐼𝑐 |

Hình 2.3 Hình ảnh trước và sau khi thực hiện phép dò biên Canny

## Thuật toán tối ưu động vật biển ăn thịt MPA

### Ý tưởng chính

Thuật toán tối ưu động vật biển ăn thịt MPA (Marine Predators Algorithm) được đề xuất bởi A. Faramarzi và các đồng nghiệp [1], được lấy cảm hứng từ hành vi săn mồi của các loài động vật sống dưới nước như cá mập, cá voi. Thuật toán này được sử dụng để giải quyết các bài toán tối ưu hóa trong lĩnh vực kỹ thuật và khoa học, như tối ưu hóa các hàm số, mô hình hóa và dự đoán, và điều khiển hệ thống. Cơ chế của thuật toán MPA được lấy cảm hứng từ cách các loài động vật săn mồi tìm kiếm và đuổi theo con mồi trong nước. Thuật toán MPA bao gồm một quần thể các "cá thể" ảo, mỗi cá thể được mô tả bằng một vị trí trong không gian tìm kiếm. Trong quá trình tối ưu hóa, các cá thể sẽ di chuyển và tương tác với nhau dựa trên một số luật và quy tắc của hành vi săn mồi của các loài động vật. Cụ thể, các cá thể trong quần thể sẽ được phân vào hai loại: cá thể "săn mồi" và cá thể "cạnh tranh". Các cá thể săn mồi sẽ cố gắng di chuyển tới các vị trí có giá trị tối ưu hơn, trong khi các cá thể cạnh tranh sẽ cố gắng cản trở và giảm hiệu quả của các cá thể săn mồi. Thuật toán MPA đã cho thấy hiệu quả trong việc giải quyết các bài toán tối ưu hóa trong nhiều lĩnh vực, như tối ưu hóa các mô hình hóa và dự đoán, tối ưu hóa các hàm số, và điều khiển các hệ thống.

### Các bước giải thuật

**Bước 1:** Chọn thông số đầu vào gồm:

•  𝑁: Số lượng cá thể trong quần thể

• : Số lần lặp để tìm kiếm lời giải tối ưu

• giá trị cận trên của con mồi

•  : giá trị cận dưới của con mồi

•  : Số lượng biến trong bài toán tối ưu

•  K: là hằng số mang giá trị 0.5

•  FADs: là tham số mang giá trị 0.2

**Bước 2:** Khởi tạo vị trí cá thể trong quần thể:

Vị trí của các cá thể là ngẫu nhiên: Prey(x1, x2)

**Bước 3:** Bắt đầu vòng lặp:

Xác định ma trận đầu vào:

là ma trận được xây dựng dựa trên nghiệm phù hợp nhất. Trong vòng đầu tiên ma trận E sẽ được khởi tạo ngẫu nhiên.

là vector chứa các giá trị ngẫu nhiên tuân theo phân phối đồng đều.

là vector chứa các số ngẫu nhiên được sinh ra theo phân phối chuẩn.

là vector được tạo bơi phân phối Lévy.

Là vector nhị phân

**Bước 4:** Nếu vòng lặp iter <max\_iter thì kích thước bước di chuyển của con mồi () và vị trí của nó () được xác định bởi các phương trình (2-15) và

(2-16).

|  |  |
| --- | --- |
|  | (2‑15) |
|  | (2‑16) |

Trong đó:

* là vector chứa kích thước bước di chuyển của con mồi.
* là ma trận được xây dựng dựa trên nghiệm phù hợp nhất.
* là vector chứa các giá trị ngẫu nhiên tuân theo phân phối đồng đều.
* là hằng số mang giá trị 0.5.
* là toán tử nhân từng phần tử của 2 ma trận.
* là vector chứa các số ngẫu nhiên được sinh ra theo phân phối chuẩn.

Nếu vòng lặp iter > max\_iter và iter > max\_iter thì kích thước bước di chuyển của con mồi và vị trí của nó được xác định bởi các phương trình (2-17), (2-18), (2-19) và (2-20).

Đối với nửa đầu số con mồi:

|  |  |
| --- | --- |
|  | (2‑17) |
|  | (2‑18) |

Đối với nửa số con mồi còn lại:

|  |  |
| --- | --- |
|  | (2‑19) |
|  | (2‑20) |

Trong đó

Nếu thì kích thước bước di chuyển của con mồi và vị trí của nó được xác định bởi các phương trình (2-21) và (2-22).

|  |  |
| --- | --- |
|  | (2‑21) |
|  | (2‑22) |

Trong đó

* là vector được tạo bơi phân phối Lévy.

**Bước 5:** Chọn nghiệm tốt Top\_predator nhất bằng cách thay các giá trị vào hàm 𝑓 (x) cần tối ưu. Nghiệm tốt nhất sẽ cho giá trị tốt nhất

**Bước 6:** Tạo cá thể con mồi tiếp theo:

Xác định kích thước bước di chuyển và vị trí con mồi khi hình thành xoáy và tác động bởi FADs:

Nếu FADs r vị trí của con mồi sẽ được xác định bởi phương trình (2-23).

|  |  |
| --- | --- |
| ) | (2‑23) |

Nếu FADs r vị trí của con mồi sẽ được xác định bởi phương trình (2-24).

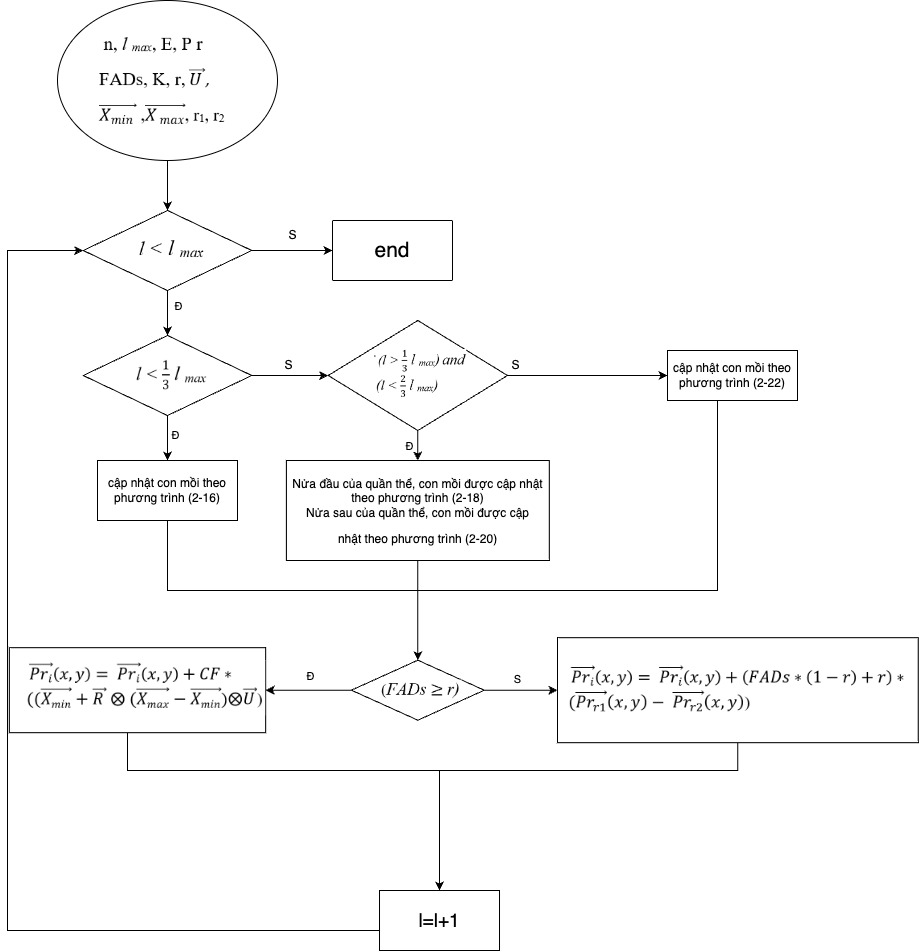
|  |  |
| --- | --- |
| ) | (2‑24) |

**Bước 7:** Kiểm tra điều kiện dừng vòng lặp: iter < max\_𝑖𝑡𝑒𝑟

Nếu thoả mãn tiếp tục vòng lặp tại bước 3.

Nếu không thỏa mãn, thì kết thúc vòng lặp lấy vị trí con mồi tốt nhất.

### Sơ đồ thuật toán MPA



Hình 2.4. Sơ đồ thuật toán MPA

### Ví dụ minh họa

Tối ưu hóa phương trình: 𝑓 = (x1 – 2)2 + (x2 – 1)2 + 3

**Bước 1:** Chọn thông số đầu vào gồm:

•  Số lượng cá thể trong quần thể

• Số lần lặp để tìm kiếm lời giải tối ưu

•  Giá trị cận trên của con mồi

•  Giá trị cận dưới của con mồi

•  Số lượng biến trong bài toán tối ưu

Hằng số mang giá trị

•  Tham số ngẫu nhiên

**Bước 2:** Khởi tạo vị trí cá thể trong quần thể:

Vị trí của các cá thể là ngẫu nhiên:

|  |  |  |
| --- | --- | --- |
|  | 1.8771 | 3.9940 |
| Prey = | 2.3032 | 3.1732 |
|  | 1.3838 | 2.3525 |
|  | 3.9901 | 4.3293 |

Kiểm tra thỏa mãn điều kiện vòng lặp

**Bước 3:** Bắt đầu vòng lặp (vòng lặp 1):

|  |  |  |
| --- | --- | --- |
|  | 1.8771 | 3.9940 |
| Prey = | 2.3032 | 3.1732 |
|  | 1.3838 | 2.3525 |
|  | 3.9901 | 4.3293 |

Xác định vị trí tốt nhất của phương trình tối ưu:

|  |  |
| --- | --- |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (1.8771– 2)2 + (3.9940 – 1)2 + 3 |
|  | = 11.9792 |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (2.3032 – 2)2 + (3.1732 – 1)2 + 3 |
|  | = 7.8147 |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (1.3838 – 2)2 + (2.3525 – 1)2 + 3 |
|  | = 5.2090 |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (3.9901– 2)2 + (4.3293 – 1)2 + 3 |
|  | = 18.0447 |

và

Xác định ma trận đầu vào:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1.3838 | 2.3525 |  |  | 0.0046 | 0.0112 |
| = | 1.3838 | 2.3525 |  | = | 0.0421 | -0.0457 |
|  | 1.3838 | 2.3525 |  |  | 0.0741 | 0.0313 |
|  | 1.3838 | 2.3525 |  |  | 0.0046 | 0.0112 |

|  |  |  |
| --- | --- | --- |
|  | 0.6067 |  |
| = | 1.6345 |  |
|  |  |  |
|  |  |  |

**Bước 4:** Do và max\_iter =5 nên

Vì vậy kích thước bước di chuyển của con mồi () và vị trí của nó () được xác định bởi các phương trình (2-15) và (2-16).

|  |  |
| --- | --- |
|  | (2-15) |
|  | (2-16) |

|  |  |
| --- | --- |
| 1,1] | = |
|  | = (0.6067) \* (1.3838) (0.6067) (1.8771) |
|  | = 0.1486 |

Số ngẫu nhiên R = 0.0513

|  |  |
| --- | --- |
| Prey[1,1] | = |
|  | = (1.8771) + (0.5) \*(0.0513) \*(0.1486) |
|  | = 1.8809 |
| 1,2] | = |
|  | = |
|  |  |

Số ngẫu nhiên R = 0.5927

|  |  |
| --- | --- |
| Prey[1,2] | = |
|  |  |
|  | = 1.5850 |
| 2,1] | = |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.1629

|  |  |
| --- | --- |
| Prey[2,1] | = |
|  | = |
|  | = 1.9863 |
| 2,2] | = |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.8384

|  |  |
| --- | --- |
| Prey[2,2] | = |
|  |  |
|  | = 1.0559 |
| 3,1] | = |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.1676

|  |  |
| --- | --- |
| Prey[3,1] | = |
|  | = |
|  | = 1.2664 |
| 3,2] | = |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.5022

|  |  |
| --- | --- |
| Prey[3,2] |  |
|  |  |
|  |  |
| 4,1] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.9993

|  |  |
| --- | --- |
| Prey[4,1] |  |
|  |  |
|  |  |

Vì

|  |  |
| --- | --- |
| 4,2] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.3554

|  |  |
| --- | --- |
| Prey[4,2] |  |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
|  | 1.8809 | 1.585 |
| Prey = | 1.9863 | 1.0559 |
|  | 1.2664 | 1.6899 |
|  | 0 | 4.3145 |

Bước 5: Chọn nghiệm tốt nhất bằng cách thay các giá trị vào hàm 𝑓 (x) cần tối ưu. Nghiệm tốt nhất sẽ cho giá trị tốt nhất

Kết quả vòng lặp 1:

Xác định vị trí tốt nhất của phương trình tối ưu:

|  |  |
| --- | --- |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (1.8809– 2)2 + (1.585– 1)2 + 3 |
|  | = 3.3564 |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (1.9863 – 2)2 + (1.0559– 1)2 + 3 |
|  | = 3.0033 |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (1.2664– 2)2 + (1.6899– 1)2 + 3 |
|  | = 4.0141 |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (0– 2)2 + (4.3145– 1)2 + 3 |
|  | = 17.9859 |

Vì

=> và

**Bước 6:** Tạo cá thể con mồi tiếp theo:

Xác định kích thước bước di chuyển và vị trí con mồi khi hình thành xoáy và tác động bởi FADs:

r = 0.0471

CF = 1

|  |  |  |
| --- | --- | --- |
|  | 0 | 0 |
| U = | 0 | 0 |
|  | 0 | 0 |
|  | 0 | 0 |

FADs r vị trí của con mồi sẽ được xác định bởi phương trình (2-23).

|  |  |
| --- | --- |
| ) | (2-23) |
| Suy ra: |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Prey = |  |  |
|  |  |  |
|  |  |  |

**Bước 7:** Tăng

Kiểm tra điều kiện dừng vòng lặp: thỏa mãn điều kiện vòng lặp

Tiếp tục vòng lặp tại bước 3.

**Bước 3:** Bắt đầu vòng lặp (vòng lặp 2):

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Prey = |  |  |
|  |  |  |
|  |  |  |

Xác định vị trí tốt nhất của phương trình tối ưu:

|  |  |
| --- | --- |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (1.8809 – 2)2 + (1.585 – 1)2 + 3 |
|  | = 3.3564 |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (1.9863 – 2)2 + (1.0559 – 1)2 + 3 |
|  | = 3.0033 |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (1.2664 – 2)2 + (1.6899 – 1)2 + 3 |
|  | = 4.0141 |
| 𝑓 (x1, x2) | = (x1 – 2)2 + (x2 – 1)2 + 3= (0– 2)2 + (4.3145 – 1)2 + 3 |
|  | = 17.9859 |

Vì

=> và

Xác định ma trận đầu vào:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
| = |  |  |  | = |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| = |  |  |
|  |  |  |
|  |  |  |

**Bước 4:** Dovà nên

Vì vậy kích thước bước di chuyển của con mồi () và vị trí của nó () được xác định bởi các phương trình (2-15) và (2-16).

|  |  |  |
| --- | --- | --- |
|  | | (2-15) |
|  | | (2-16) |
| 1,1] |  | |
|  |  | |
|  |  | |

Số ngẫu nhiên R = 0.7008

|  |  |
| --- | --- |
| Prey[1,1] |  |
|  |  |
|  |  |
| 1,2] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.7425

|  |  |
| --- | --- |
| Prey[1,2] |  |
|  |  |
|  | 1.5339 |
| 2,1] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.7579

|  |  |
| --- | --- |
| Prey[2,1] |  |
|  |  |
|  | 1.6104 |
| 2,2] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.3891

|  |  |
| --- | --- |
| Prey[2,2] |  |
|  |  |
|  | 0.8812 |
| 3,1] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.4293

|  |  |
| --- | --- |
| Prey[3,1] |  |
|  |  |
|  |  |

Vì

|  |  |
| --- | --- |
| 3,2] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.9563

|  |  |
| --- | --- |
| Prey[3,2] |  |
|  |  |
|  | 1.6134 |
| 4,1] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.5730

|  |  |
| --- | --- |
| Prey[4,1] |  |
|  |  |
|  |  |
| 4,2] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.8497

|  |  |
| --- | --- |
| Prey[4,2] |  |
|  |  |
|  | 4.2844 |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| =>Prey = |  |  |
|  |  |  |
|  |  |  |

**Bước 5:** Chọn nghiệm tốt **Top\_predator** nhất bằng cách thay các giá trị vào hàm 𝑓 (x) cần tối ưu. Nghiệm tốt nhất sẽ cho giá trị tốt nhất

Kết quả vòng lặp 2:

Xác định vị trí tốt nhất của phương trình tối ưu:

|  |  |
| --- | --- |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (2.0627– 2)2 + (1.5339– 1)2 + 3 |
|  | 3.2890 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.6104– 2)2 + (0.8812– 1)2 + 3 |
|  | 3.1659 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (0– 2)2 + (1.6134 – 1)2 + 3 |
|  |  |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (– 2)2 + (4.2844– 1)2 + 3 |
|  | 11.8311 |

Vì

=> và

**Bước 6:** Tạo cá thể con mồi tiếp theo:

Xác định kích thước bước di chuyển và vị trí con mồi khi hình thành xoáy và tác động bởi FADs:

r = 0.2763

CF = 0.9146

FADs r vị trí của con mồi sẽ được xác định bởi phương trình (2-24).

|  |  |
| --- | --- |
| ) | (2-24) |

Suy ra:

|  |  |  |
| --- | --- | --- |
|  | 1.9505 | 2.6264 |
| Prey = | 1.9541 | 1 |
|  | 1.2986 | 1.8912 |
|  | 1.1122 | 3.1919 |

**Bước 7:** Tăng

Kiểm tra điều kiện dừng vòng lặp: thỏa mãn điều kiện vòng lặp

Tiếp tục vòng lặp tại bước 3

**Bước 3:** Bắt đầu vòng lặp (vòng lặp 3):

|  |  |  |
| --- | --- | --- |
|  | 1.9505 | 2.6264 |
| Prey = | 1.9541 | 1 |
|  | 1.2986 | 1.8912 |
|  | 1.1122 | 3.1919 |

Xác định vị trí tốt nhất của phương trình tối ưu:

|  |  |
| --- | --- |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.9505– 2)2 + (2.6264– 1)2 + 3 |
|  | 5.6476 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.9541– 2)2 + (1– 1)2 + 3 |
|  | 3.0021 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1– 2)2 + (1.2986– 1)2 + 3 |
|  | 4.0891 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.1122– 2)2 + (3.1919– 1)2 + 3 |
|  | 8.5926 |

Vì

=>và

Xác định ma trận đầu vào:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
| = |  |  |  | = |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  | - |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| = |  |  |
|  |  |  |
|  |  |  |

Vì vậy kích thước bước di chuyển của con mồi và vị trí của nó được xác định bởi các phương trình (2-17), (2-18), (2-19) và (2-20).

Đối với nửa đầu số con mồi:

|  |  |
| --- | --- |
|  | (2-17) |
|  | (2-18) |

Đối với nửa số con mồi còn lại:

|  |  |
| --- | --- |
|  | (2-19) |
|  | (2-20) |

**Nửa đầu:**

|  |  |
| --- | --- |
| 1,1] |  |
|  |  |
|  | 1.7409 |

Số ngẫu nhiên R = 0.9852

|  |  |
| --- | --- |
| Prey[1,1] |  |
|  |  |
|  | 2.8081 |
| 1,2] |  |
|  |  |
|  | 0.0102 |

Số ngẫu nhiên R = 0.5595

|  |  |
| --- | --- |
| Prey[1,2] |  |
|  |  |
|  | 2.6293 |
| 2,1] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.9336

|  |  |
| --- | --- |
| Prey[2,1] |  |
|  |  |
|  | 1.9433 |
| 2,2] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.7203

|  |  |
| --- | --- |
| Prey[2,2] |  |
|  |  |
|  | 0.9970 |

**Nửa sau:**

CF = 0.6645

|  |  |
| --- | --- |
| 3,1] |  |
|  |  |
|  | 5.1730 |
| Prey[3,1] |  |
|  |  |
|  | 3.6728 |
| 3,2] |  |
|  |  |
|  | 0.7544 |
| Prey[3,2] |  |
|  |  |
|  | 1.2506 |
| 4,1] |  |
|  |  |
|  | 3.4100 |
| Prey[4,1] |  |
|  |  |
|  | 3.0871 |
| 4,2] |  |
|  |  |
|  |  |
| Prey[4,2] |  |
|  |  |
|  | 0.3839 |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Prey = |  |  |
|  |  |  |
|  |  |  |

**Bước 5:** Chọn nghiệm tốt **Top\_predator** nhất bằng cách thay các giá trị vào hàm 𝑓 (x) cần tối ưu. Nghiệm tốt nhất sẽ cho giá trị tốt nhất

Kết quả vòng lặp 3:

Xác định vị trí tốt nhất của phương trình tối ưu:

|  |  |
| --- | --- |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (2.8081 – 2)2 + (2.6293 – 1)2 + 3 |
|  | 6.3076 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.9433– 2)2 + (0.9970– 1)2 + 3 |
|  | 3.0032 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (3.6728– 2)2 + (1.2506– 1)2 + 3 |
|  | 5.8611 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (3.0871– 2)2 + (0.33839– 1)2 + 3 |
|  | 4.5613 |

Vì

=> và

**Bước 6:** Tạo cá thể con mồi tiếp theo:

Xác định kích thước bước di chuyển và vị trí con mồi khi hình thành xoáy và tác động bởi FADs:

r = 0.3953

FADs r vị trí của con mồi sẽ được xác định bởi phương trình (2-24).

|  |  |
| --- | --- |
|  | (2-24) |

Suy ra:

|  |  |  |
| --- | --- | --- |
|  | 1.4778 | 1.5339 |
| Prey = | 1.9541 | 1 |
|  | 1.8514 | 1.6899 |
|  | 3.0871 | 1 |

**Bước 7:** Tăng

Kiểm tra điều kiện dừng vòng lặp:thỏa mãn điều kiện vòng lặp

Tiếp tục vòng lặp tại bước 3

**Bước 3:** Bắt đầu vòng lặp (vòng lặp 4):

|  |  |  |
| --- | --- | --- |
|  | 1.4778 | 1.5339 |
| Prey = | 1.9541 | 1 |
|  | 1.8514 | 1.6899 |
|  | 3.0871 | 1 |

Xác định vị trí tốt nhất của phương trình tối ưu:

|  |  |
| --- | --- |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.4778 – 2)2 + (1.5339 – 1)2 + 3 |
|  | 3.5577 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.9541– 2)2 + (1– 1)2 + 3 |
|  | 3.0021 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1– 2)2 + (1.8514– 1.6899)2 + 3 |
|  | 4.0261 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (3.0871 – 2)2 + (1 – 1)2 + 3 |
|  | 4.1818 |

Vì

=> và

Xác định ma trận đầu vào:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1.9541 | 1 |  |  | 0.0681 | 0.0252 |
| = | 1.9541 | 1 |  | = | 0.0067 | 0.0046 |
|  | 1.9541 | 1 |  |  | 0.1194 |  |
|  | 1.9541 | 1 |  |  | 0.2230 | 0.0532 |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| = |  |  |
|  |  |  |
|  |  |  |

**Bước 4:** Vì và nên

Vì vậy kích thước bước di chuyển của con mồi và vị trí của nó được xác định bởi các phương trình (2-17), (2-18), (2-19) và (2-20).

Đối với nửa đầu số con mồi:

|  |  |
| --- | --- |
|  | (2-17) |
|  | (2-18) |

Đối với nửa số con mồi còn lại:

|  |  |
| --- | --- |
|  | (2-19) |
|  | (2-20) |

**Nửa đầu:**

|  |  |
| --- | --- |
| 1,1] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.4443

|  |  |
| --- | --- |
| Prey[1,1] |  |
|  |  |
|  | 1.5058 |
| 1,2) |  |
|  |  |
|  | 0.0242 |

Số ngẫu nhiên R = 0.7559

|  |  |
| --- | --- |
| Prey[1,2] |  |
|  |  |
|  | 1.5430 |
| 2,1] |  |
|  |  |
|  | 0.0130 |

Số ngẫu nhiên R = 0.6032

|  |  |
| --- | --- |
| Prey[2,1] |  |
|  |  |
|  | 1.9580 |
| 2,2] |  |
|  |  |
|  |  |

Số ngẫu nhiên R = 0.7833

|  |  |
| --- | --- |
| Prey[2,2] |  |
|  |  |
|  | 1.0018 |

**Nửa sau:**

CF = 0.3330

|  |  |
| --- | --- |
| 3,1] |  |
|  |  |
|  | 1.9294 |
| Prey[3,1] |  |
|  |  |
|  | 2.2753 |
| 3,2] |  |
|  |  |
|  |  |
| Prey[3,2] |  |
|  |  |
|  | 1.8006 |
| 4,1] |  |
|  |  |
|  |  |
| Prey[4,1] |  |
|  |  |
|  | 1.7909 |
| 4,2] |  |
|  |  |
|  | 1.4653 |
| Prey[4,2] |  |
|  |  |
|  | 1.2440 |

|  |  |  |
| --- | --- | --- |
|  | 1.5058 | 1.5430 |
| =>Prey = | 1.9580 | 1.0018 |
|  | 2.2753 | 1.8006 |
|  | 1.7909 | 1.2440 |

**Bước 5:** Chọn nghiệm tốt Top\_predator nhất bằng cách thay các giá trị vào hàm 𝑓 (x) cần tối ưu. Nghiệm tốt nhất sẽ cho giá trị tốt nhất

Kết quả vòng lặp 4:

Xác định vị trí tốt nhất của phương trình tối ưu:

|  |  |
| --- | --- |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.5058 – 2)2 + (1.5430 – 1)2 + 3 |
|  | 3.5391 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.9580 – 2)2 + (1.0018– 1)2 + 3 |
|  | 3.0018 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (2.2753– 2)2 + (1.8006– 1)2 + 3 |
|  | 3.7168 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.7909 – 2)2 + (1.2440 – 1)2 + 3 |
|  | 3.1033 |

Vì

=> và

**Bước 6:** Tạo cá thể con mồi tiếp theo:

Xác định kích thước bước di chuyển và vị trí con mồi khi hình thành xoáy và tác động bởi FADs:

r = 0.4662

FADs r vị trí của con mồi sẽ được xác định bởi phương trình (2-24).

|  |  |
| --- | --- |
| ) | (2-24) |

Suy ra:

|  |  |  |
| --- | --- | --- |
|  | 2.0974 | 1.7894 |
| Prey = | 2.0791 | 1 |
|  | 1.7556 | 1.8287 |
|  | 1.7308 | 1 |

**Bước 7:** Tăng

Kiểm tra điều kiện dừng vòng lặp: thỏa mãn điều kiện vòng lặp

Tiếp tục vòng lặp tại bước 3

**Bước 3:** Bắt đầu vòng lặp (vòng lặp 5):

|  |  |  |
| --- | --- | --- |
|  | 2.0974 | 1.7894 |
| Prey = | 2.0791 | 1 |
|  | 1.7556 | 1.8287 |
|  | 1.7308 | 1 |

Xác định vị trí tốt nhất của phương trình tối ưu:

|  |  |
| --- | --- |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (2.0974 – 2)2 + (1.7894 – 1)2 + 3 |
|  | 3.6326 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (2.0791 – 2)2 + (1 – 1)2 + 3 |
|  | 3.0063 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.7556 – 2)2 + (1.8287 – 1.6899)2 + 3 |
|  | 3.0790 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.7308 – 2)2 + (1 – 1)2 + 3 |
|  | 3.0725 |

Vì

=> và

Xác định ma trận đầu vào:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
| = |  |  |  | = |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| = |  |  |
|  |  |  |
|  |  |  |

**Bước 4:**

Vì vậy kích thước bước di chuyển của con mồi và vị trí của nó được

xác định bởi các phương trình (2-21) và (2-22).

|  |  |
| --- | --- |
|  | (2-21) |
|  | (2-22) |

CF = 0.0761

|  |  |
| --- | --- |
| 1,1] |  |
|  |  |
|  |  |
| Prey[1,1] |  |
|  |  |
|  | 1.9579 |
| 1,2] |  |
|  |  |
|  | 0.01059 |
| Prey[1,2] |  |
|  |  |
|  | 1.0022 |
| 2,1] |  |
|  |  |
|  | 0.0476 |
| Prey[2,1] |  |
|  |  |
|  | 1.9598 |
| 2,2] |  |
|  |  |
|  |  |
| Prey[2,2] |  |
|  | (1.0018) + (0.5) \*(0.0761) \*(-0.0056) |
|  | 1.0016 |
| 3,1] |  |
|  |  |
|  | 0.0909 |
| Prey[3,1] |  |
|  |  |
|  | 1.9615 |
| 3,2] |  |
|  |  |
|  |  |
| Prey[3,2] |  |
|  |  |
|  | 0.9929 |
| 4,1] |  |
|  |  |
|  | 0.0164 |
| Prey[4,1] |  |
|  |  |
|  | 1.9586 |
| 4,2] |  |
|  |  |
|  |  |
| Prey[4,2] |  |
|  |  |
|  | 1.0008 |

|  |  |  |
| --- | --- | --- |
|  | 1.9579 | 1.0022 |
| =>Prey = | 1.9598 | 1.0016 |
|  | 1.9615 | 0.9929 |
|  | 1.9586 | 1.0008 |

**Bước 5:** Chọn nghiệm tốt **Top\_predator** nhất bằng cách thay các giá trị vào hàm 𝑓 (x) cần tối ưu. Nghiệm tốt nhất sẽ cho giá trị tốt nhất

Kết quả vòng lặp 5:

Xác định vị trí tốt nhất của phương trình tối ưu:

|  |  |
| --- | --- |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.9579 – 2)2 + (1.0022 – 1)2 + 3 |
|  | 3.0017 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.9598 – 2)2 + (1.0016 – 1)2 + 3 |
|  | 3.0016 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.9615 – 2)2 + (0.9929 – 1)2 + 3 |
|  | 3.0015 |
| 𝑓 (x1, x2) | (x1 – 2)2 + (x2 – 1)2 + 3= (1.9586 – 2)2 + (1.0008– 1)2 + 3 |
|  | 3.0017 |

Vì

=> và

**Bước 6:** Tạo cá thể con mồi tiếp theo:

Xác định kích thước bước di chuyển và vị trí con mồi khi hình thành xoáy và tác động bởi FADs:

r = 0.8776

FADs r vị trí của con mồi sẽ được xác định bởi phương trình (2-24).

|  |  |
| --- | --- |
| ) | (2-24) |

Suy ra:

|  |  |  |
| --- | --- | --- |
|  | 1.9579 | 1.0021 |
| Prey = | 1.9615 | 1.0002 |
|  | 1.9599 | 1.0014 |
|  | 1.9586 | 1.0008 |

**Bước 7:** Tăng

Kiểm tra điều kiện dừng vòng lặp: Không thỏa mãn điều kiện vòng lặp

**Kết thúc vòng lặp**

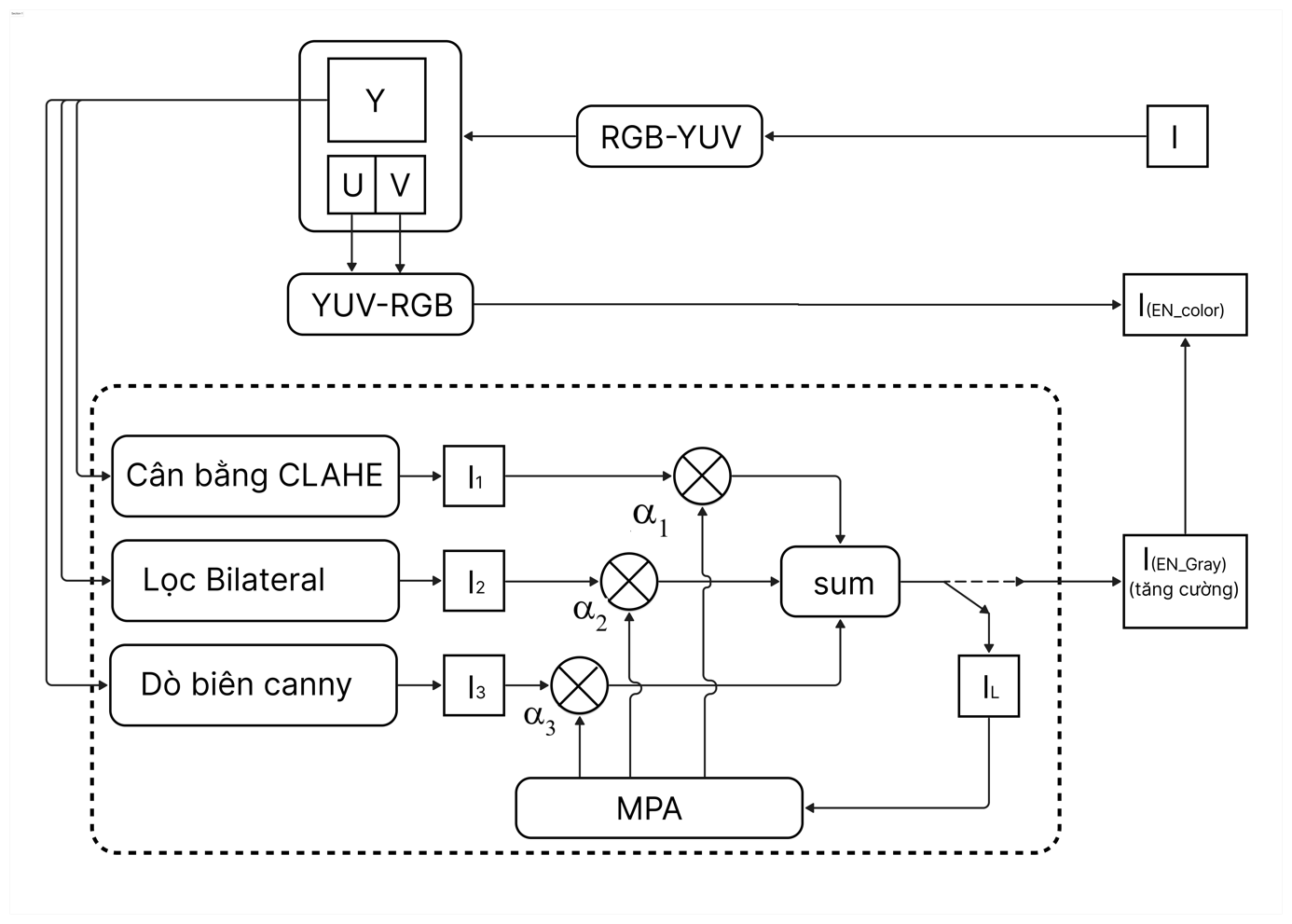
Kết luận: Vậy vị trí con mồi tốt nhất và giá trị hàm số tối ưu tốt nhất là .

# ĐỀ XUẤT PHƯƠNG PHÁP TĂNG CƯỜNG CHẤT LƯỢNG ẢNH

## Mô hình đề xuất

Đầu vào: Ảnh màu I thiếu sáng.

Đầu ra: Hình ảnh nâng cao IEN\_Color



Hình 3.1 Các bước của kỹ thuật tăng cường ảnh

Các bước thực hiện của giải thuật đề xuất như sau:

**Bước 1**: Hình ảnh đầu vào I được chuyển đổi sang không gian màu YUV, thu được các kênh Y, U và V. Tách kênh độ sáng Y để xử lý tiếp theo, giữ lại kênh U và kênh V.

**Bước 2**: Thành phần I1 được tăng cường độ sáng bằng phương pháp cân bằng CLAHE

**Bước 3**: Thành phần I2 được khử nhiễu bằng phương pháp lọc Biratelar

**Bước 4**: Thành phần I3 được làm mịn bằng phương pháp dò biên Canny

**Bước 5:** Áp dụng thuật toán tối ưu MPA để tìm 3 tham số tối ưu là vàHàm mục tiêu được mô tả là phương trình (3-1).

|  |  |
| --- | --- |
|  | (3‑1) |
|  | (3‑2) |
|  | (3‑3) |

Trong đó:

* V là phương sai của ảnh
* M là cường độ sáng trung bình của ảnh
* lần lượt là entropy của ảnh Y và
* là hình ảnh được tăng cường theo thời gian được tính toán trong mỗi lần lặp lại thuật toán MPA dưới dạng biểu thức (3-2).

|  |  |
| --- | --- |
|  | (3‑4) |

|  |  |
| --- | --- |
|  | (3‑5) |

**Bước 7:** Ba thành phần U và V được chuyển đổi sang không gian màu RGB, .

## Thực nghiệm và đánh giá

### Môi trường thực nghiệm

Ngôn ngữ lập trình: Matlab.

Cấu hình máy tính:

* CPU: 8-core **CPU**/8GB/256GB/7-core GPU (MGN63SA/A)
* RAM: 8GB.
* Chip: M1

### Dữ liệu thực nghiệm

Dữ liệu thử nghiệm được lấy từ Bộ dữ liệu nguồn LOL[[4]](#footnote-4). Trong nghiên cứu này sẽ sử dụng 30 ảnh từ A1 đến A30 đã liệt kê tại mục 1.2 để thực hiện thực nhiệm.

### Thiết kế thực nghiệm

**Thực nghiệm 1:** Để kiếm tra sự hiệu quả của thuật toán MPA trong phương pháp tăng cường ảnh đề xuất, một số giải thuật tối ưu khác được sử dụng để so sánh minh hoạ trong bảng 3-1. Dữ liệu được sử dụng trong thực nghiệm này là hình ảnh A3 để thực hiện thực nghiệm đánh giá hiệu suất của giải thuật đề xuất và một số giải thuật tối ưu khác.

Mỗi giải thuật tối ưu được chạy 30 lần khác nhau trên ảnh A3.

Hiệu quả của thuật toán tối ưu được đánh giá dựa trên 2 chỉ số đánh giá:

* Giá trị trung bình
* Độ lệch chuẩn

Bảng: 3‑1 Một số giải thuật tối ưu được sử dụng để so sánh

|  |  |  |
| --- | --- | --- |
| STT | Thuật toán | Tên đầy đủ |
| 1 | GOA [25] | Grasshopper Optimization Algorithm |
| 2 | MRFO [39] | Maximal-Rank Fixed-Point Optimization |
| 3 | WOA [40] | Whale Optimization Algorithm |
| 4 | ALO [41] | Ant Lion Optimizer |

**Thực nghiệm 2:** Để kiểm tra hiệu quả của phương pháp đề xuất, một số giải thuật nâng cao chất lượng ảnh được so sánh minh hoạ trong bảng 3-2. Bộ dữ liệu được sử dụng trong thực nghiệm này gồm 30 ảnh, từ A1- A30, được dùng để so sánh chất lượng ảnh sau khi áp dụng phương pháp tăng cường ảnh bằng giải thuật đề xuấtvới chất lượng ảnh sau khi áp dụng một số giải thuật tăng cường chất lượng ảnh tối khác.

Bảng: 3‑2 Một số giải thuật nâng cao chất lượng ảnh được so sánh

|  |  |  |
| --- | --- | --- |
| STT | Thuật toán | Tên đầy đủ |
| 1 | NE | No enhancement |
| 2 | HE | Histogram equalization |
| 3 | CLAHE [2] | Contrast Limited Adaptive Histogram Equalization |
| 4 | EGIF [42] | Effective guided image filtering |

Hiệu quả của thuật toán tối ưu được đánh giá dựa trên 4 chỉ số đánh giá:

* Kết quả đo độ sáng (M),
* Độ tương phản (CL),
* Lượng thông tin entropy (E),
* Độ sắc nét (G)

## Kết quả thực nghiệm

**Thực nghiệm 1:**

Trong thực nhiệm này, mỗi thuật toán tối ưu được chạy 30 lần khác nhau và đánh giá bằng cách sử dụng 2 chỉ số: Giá trị trung bình (mean) và độ lệch chuẩn (Std) cho các giá trị của hàm Fitness thu được.

Bảng 3-3 mô tả chỉ số trung bình của hàm Fitness và độ lệch chuẩn của các giải thuật tối ưu từ đó đưa ra đánh giá của các giải thuật.

Một số tham số khởi tạo trong thực nghiệm này như sau:

* Ảnh đầu vào: A3
* Số lượng cá thể: n = 25.
* Max\_iter = 30.
* Số lượng biến cần tối ưu: dim = 3
* Cận trên: ub = [0.01, 0.01, 0.01]
* Cận dưới: lb = [0.5, 0.5, 0.5]

Bảng: 3‑3 Giá trị trung bình và độ lệch chuẩn của các giải thuật so sánh

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Thuật toán | Trung bình | Độ lệch chuẩn |
| 1 | GOA |  |  |
| 2 | MRFO |  |  |
| 3 | WOA |  |  |
| 4 | ALO |  |  |
| 5 | Đề xuất |  |  |

Kết quả thực nghiệm cho thấy các chỉ số đánh giá thu được từ giải thuật MPA có giá trị trung bình cao nhất hay hàm tối ưu đạt kết quả tốt nhất và độ lệch chuẩn là thấp nhất từ đó giải thích vì sao giải thuật tối ưu MPA được sử dụng trong mô hình đề xuất.

**Thực nghiệm 2:**

Trong thực nhiệm 2, mỗi giải thuật nâng cao chất lượng ảnh được chạy với bộ dữ liệu gồm 30 ảnh tối từ A1 đến A30. Các chỉ số đánh giá: độ sáng (M), độ tương phản (CL), lượng thông tin entropy (E), độ sắc nét (G).

Một số tham số khởi tạo trong thực nghiệm này:

* Ảnh đầu vào: A1 đến A30
* Số lượng cá thể: n = 25.
* Max\_iter = 50.
* Số lượng biến cần tối ưu: dim = 3
* Cận trên: ub = [0.01, 0.01, 0.01]
* Cận dưới: lb = [0.5, 0.5, 0.5]

Các hình ảnh 3.2, 3.3, 3.4 mô tả ba hình ảnh ví dụ khi thực hiện nâng cao chất lượng ảnh tối dự trên giải thuật đề xuất cho ảnh ba ảnh: A1, A2 và A3 với ảnh đầu vào là ảnh tối và ảnh đầu ra là ảnh đã được nâng cao chất lượng.

Trong đó: , ,, là các tham số tối ưu và các chỉ số đánh giá: độ sáng (M), độ tương phản (CL), lượng thông tin entropy (E), độ sắc nét (G).

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ảnh trước tăng cường | | | | Ảnh sau khi tăng cường | | | | Tham số tối ưu |
|  | | | |  | | | | = 0.7396  = 0.5170  = 0.2000 |
| 𝜇 | 𝜎2 | E | G | 𝜇 | 𝜎2 | E | G |
| 0.2558 | 0.0537 | 6.8118 | 0.0306 | 0.4693 | 0.0555 | 7.7177 | 0.0821 |

Hình 3.2. Ảnh A1 trước và sau khi tăng cường­

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ảnh trước tăng cường | | | | Ảnh sau khi tăng cường | | | | Tham số tối ưu |
|  | | | |  | | | | = 1.000  = 0.0625  = 0.1500 |
| M | CL | E | G | M | CL | E | G |
| 0.2734 | 0.0179 | 6.9641 | 0.0529 | 0.5020 | 0.0473 | 7.7466 | 0.1321 |

Hình 3.3. Ảnh A2 trước và sau khi tăng cường

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ảnh trước tăng cường | | | | Ảnh sau khi tăng cường | | | | Tham số tối ưu |
|  | | | |  | | | | = 0.9991  = 0.0314  = 0.1532 |
| M | CL | E | G | M | CL | E | G |
| 0.4467 | 0.0456 | 7.6550 | 0.0483 | 0.5028 | 0.0646 | 7.6732 | 0.1065 |

Hình 3.4. Ảnh A3 trước và sau khi tăng cường

So sánh kết hình ảnh tăng cường trên các đề xuất tăng cường ảnh tối khác:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| NE | HE | CLAHE | EGIF | Đề xuất |

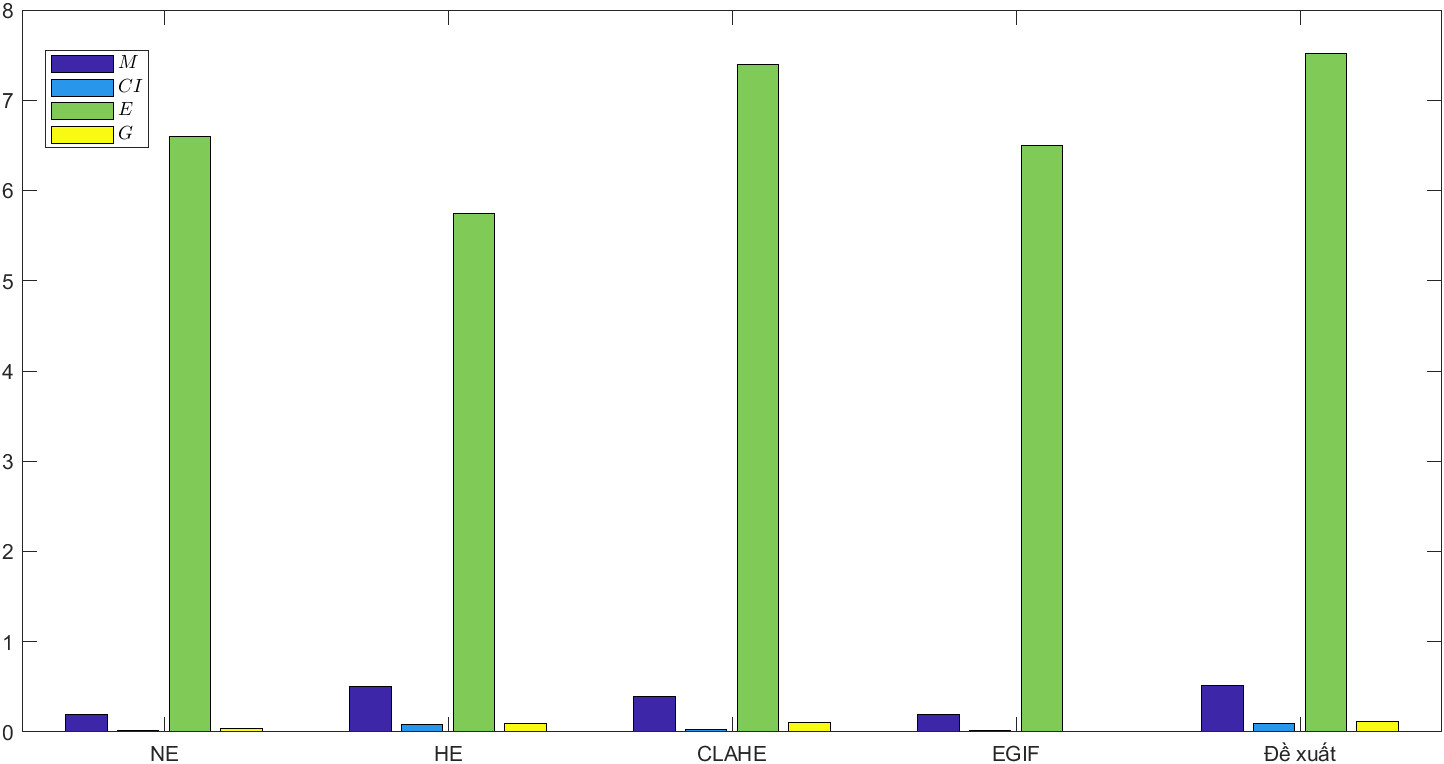
Hình 3.5. Kết quả của các phương pháp tăng cường ảnh tối (ảnh minh hoạ A1-A5)

Kết quả hình 17 cho thấy các phương pháp nâng cao ảnh đều cho kết quả tốt hơn ảnh ban đầu, độ sáng đều được cải thiện tuy nhiên, phương pháp CLAHE và HE cho ra hình ảnh có rất nhiều nhiễu. EGIF cho ra hình ảnh có độ sáng chưa được tốt.

Để thấy được sự khác biệt giữa các chỉ số của hình ảnh qua các phương pháp tăng cường ảnh dưới đây là bảng 3-4 và Hình 3.6 tổng kết thực nghiệm chạy trên 30 ảnh (A1-A30) và các chỉ số đánh giá được lấy trung bình.

Bảng: 3‑4 Giá trị trung bình các chỉ số của một số phương pháp tăng cường ảnh

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| STT | Giải thuật | M | CL | E | G |
| 1 | NE | 0.1886750 | 0.0208625 | 6.5960125 | 0.0342750 |
| 2 | HE | 0.5002083 | 0.0856250 | 5.7461916 | 0.0968425 |
| 3 | CLAHE | 0.3981583 | 0.0314083 | 7.3977625 | 0.1007400 |
| 4 | EGIF | 0.1886750 | 0.0196208 | 6.4987083 | 0.0099333 |
| 5 | Đề xuất | **0.5183083** | **0.0931391** | **7.5190043** | **0.1128695** |



Hình 3.6 Hình minh hoạ cho 4 chỉ số đánh giá của các phương pháp tăng cường ảnh khác nhau

Chỉ số đánh giá độ tương phản (), lượng thông tin entropy (E), chỉ số độ sáng (μ), độ sắc nét (G) của giải thuật đề xuất đều cao hơn các giải thuật so sánh. Những kết quả này cho thấy hình ảnh do thuật toán đề xuất tạo ra có chất lượng tốt hơn so với hình ảnh do các thuật toán nâng cao khác tạo ra.

Bảng: 3‑5 Giá trị trung bình tham số tối ưu

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 0. 9387375 | 0. 52495625 | 0. 184875 |

Các giá trị thu được từ tham số tối ưu được minh hoạ trong bảng 3-5. Giá trị   
 thu được bằng 0.938 cho phép giảm cường độ sáng của hình ảnh ( là ảnh thu được sau khi tăng cường bằng phương pháp CLAHE từ hình ảnh đầu vào I). Giá trị   
 thu được bằng 0.524 đóng góp một phần thông tin cho hình ảnh tăng cường mà không có nhiễu (Vì là ảnh được khử nhiễu từ hình ảnh đầu vào I). Giá trị   
 thu được bằng 0.184 bổ sung một phần thông tin đặc trưng (biên) cho hình ảnh tăng cường (vì là hình ảnh biên thu được thông qua phương pháp dò biên Canny).

# KẾT LUẬN

Sau khi hoàn thành đồ án, em đã đạt được các kết quả như sau:

* Tìm hiểu tổng quan về các phương pháp tăng cường ảnh.
* Nắm vững về các hệ màu RGB và YUV, cùng với cách chuyển đổi giữa các hệ màu này.
* Hiểu về các chỉ số đánh giá chất lượng ảnh màu.
* Hiểu được phương pháp tối ưu động vật biển ăn thịt (MPA).
* Đề xuất phương pháp tăng cường chất lượng ảnh tối dựa trên giải thuật tối ưu MPA và một số phương pháp tăng cường chất lượng ảnh cơ bản như: CLAHE, bộ lọc Bilateral, phương pháp phát hiện biên Canny.
* Xây dượng được chương trình nâng cao chất lượng ảnh của phương pháp đề xuất trên ngôn ngữ Matlap.

Vì thời gian có hạn và trình độ hiểu biết của em còn hạn chế nên đồ án này không thể tránh khỏi những thiếu sót. Em rất mong nhận được sự góp ý của quý thầy cô để đồ án của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | M. H. S. M. a. A. H. G. A. Faramarzi, "Marine Predators Algorithm: A nature-inspired metaheuristic," *Expert Syst. Appl,* vol. 152, p. 10.1016/j.eswa.2020.113377., 2020. |
| [2] | K. Zuiderveld, "Contrast limited adaptive histogram equalization," *Graphics gems,* pp. 474-485. |
| [3] | B. S. Rao, "Dynamic histogram equalization for contrast enhancement for digital images," *Applied Soft Computing,* no. 89, p. 106114. , 2020. |
| [4] | Z. W. Z. Z. J. L. Q. &. S. Y. Huang, "Image enhancement with the preservation of brightness and structures by employing contrast limited dynamic quadri-histogram equalization," *Optik,* no. 266, p. 165877., 2021. |
| [5] | A. B. P. &. M. S. P. Paul, "Histogram modification in adaptive bi-histogram equalization for contrast enhancement on digital images," *Optik,* no. 259, p. 168899, 2022. |
| [6] | P. K. A. S. P. R. &. A. A. Mishro, "A novel brightness preserving joint histogram equalization technique for contrast enhancement of brain MR images," *Biocybernetics and Biomedical Engineering,* no. 41(2), p. 540–553, 2021. |
| [7] | U. K. &. K. S. Acharya, "Particle swarm optimized texture based histogram equalization (Psothe) for MRI brain image enhancement," *Optik,* no. 224, p. 165760, 2020. |
| [8] | J. R. &. B. P. Jebadass, "Low light enhancement algorithm for color images using intuitionistic fuzzy sets with histogram equalization," *Multimedia Tools and Applications,* no. 81(6), p. 8093–8106, 2022. |
| [9] | D. &. N. M. K. Vijayalakshmi, "A novel contrast enhancement technique using gradient- based joint histogram equalization," *Circuits, Systems, and Signal Processing,* no. 40(8), p. 3929–3967, 2021. |
| [10] | Y. &. D. C. Huang, "Image fuzzy enhancement algorithm based on contourlet trans- form domain," *Multimedia Tools and Applications,* no. 79(47), p. 35017–35032, 2020. |
| [11] | B. D. A. &. S. A. Goyal, "An effective nonlocal means image denoising framework based on non-subsampled shearlet transform," *Soft Computing ,* p. 1–23, 2022. |
| [12] | S. R. K. R. &. R. D. S. Kollem, "Improved partial differential equation-based total variation approach to non-subsampled contourlet transform for medical image denoising," *Multime- dia Tools and Applications,* no. 80(2), p. 2663–2689, 2021. |
| [13] | M. K. F. D. M. K. O. M. C. M. K. B. &. H. A. B. Sahnoun, "A modified DWT-SVD algorithm for t1-w brain MR images contrast enhancement," *IRBM,* no. 40(4), p. 235–243, 2019. |
| [14] | L. X. F. C. S. &. L. Z. Lei, "A sparse representation denoising algorithm for finger- vein image based on dictionary learning," *Multimedia Tools and Applications,* no. 80(10), p. 15135–15159, 2021. |
| [15] | T. L. C. Z. X. &. Z. Y. Zhou, "Sparse representation with enhanced nonlocal self- similarity for image denoising," *Machine Vision and Applications,* vol. 32, no. 5, p. 1–11, 2021. |
| [16] | Z. Z. C. &. H. M. Lyu, "DSTnet: A new discrete shearlet transform-based CNN model for image denoising," *Multimedia Systems,* 2021. |
| [17] | Z. G. S. X. G. &. Z. H. Yan, "On combining CNN with non-local self-similarity based image denoising methods," *IEEE Access,* no. 8, p. 14789–14797, 2020. |
| [18] | N. L. J. &. H. Z. YU, "Fla-net: Multi-stage modular network for low-light image enhancement," *The Visual Computer,* p. 1–20, 2022. |
| [19] | X. Z. Y. M. X. Z. J. &. L. Y. Wang, "Low-light image enhancement based on gan with attention mechanism and color constancy," *Multimedia Tools and Applications,* p. 1–19, 2022. |
| [20] | M. Z. L. Z. D. N. R. L. Y. &. W. Y. Li, "Aems: an attention enhancement net- work of modules stacking for lowlight image enhancement," *The Visual Computer ,* p. 1–17, 2021. |
| [21] | P. H. inh, "An improved medical image synthesis approach based on marine predators algo- rithm and maximum gabor energy," *Neural Computing and Applications,* vol. 34, no. 6, p. 4367–4385, 2021. |
| [22] | P. H. Dinh, "A novel approach based on three-scale image decomposition and marine preda- tors algorithm for multi-modal medical image fusion," *Biomedical Signal Processing and Control,* no. 67, p. 102536, 2021. |
| [23] | P. H. Dinh, "Combining gabor energy with equilibrium optimizer algorithm for multi-modal- ity medical image fusion," *Biomedical Signal Processing and Control,* no. 68, p. 102696, 2021. |
| [24] | P. H. Dinh, "Multi-modal medical image fusion based on equilibrium optimizer algorithm and local energy functions," *Applied Intelligence,* 2021. |
| [25] | P. H. Dinh, "A novel approach based on grasshopper optimization algorithm for medical image fusion," *Expert Systems with Applications,* no. 171, p. 114576, 2021. |
| [26] | P. H. Dinh, "Combining spectral total variation with dynamic threshold neural P systems for medical image fusion," *Biomedical Signal Processing and Control,* no. 80, p. 104343, 2023. |
| [27] | V. &. I. G. Sathananthavathi, "Particle swarm optimization based retinal image enhancement," *Wireless Personal Communications,* vol. 121, no. 1, p. 543–555, 2021. |
| [28] | A. &. B. A. Wadhwa, "Contrast enhancement of MRI images using morphological transforms and PSO," *Multimedia Tools and Applications,* vol. 80, no. 14, p. 21595–21613, 2021. |
| [29] | R. A. I. B. S. K. K. N. &. S. R. Guha, "Enhancement of image contrast using selfish herd optimizer," *Multimedia Tools and Applications,* vol. 81, no. 1, p. 637–657, 2022. |
| [30] | A. D. K. R. S. G. J. &. D. S. Das, "Fitness based weighted flower pollination algorithm with mutation strategies for image enhancement," *Multimedia Tools and Applications ,* p. 1–32, 2022. |
| [31] | A. K. K. P. &. M. S. Bhandari, "Salp swarm algorithm-based optimally weighted histogram framework for image enhancement," *IEEE Transactions on Instrumentation and Measurement,* vol. 69, no. 9, p. 6807–6815, 2020. |
| [32] | N. L. Q. D. Y. &. Z. Y. Du, "Color image enhancement: A metaheuristic chimp optimization algorithm," *Neural Processing Letters,* p. 1–40, 2022. |
| [33] | L. A.-O. N. K. E. M. A. &. H. E. H. Abualigah, "Boosting marine preda- tors algorithm by salp swarm algorithm for multilevel," 2022. |
| [34] | P. H. Dinh, "A novel approach using structure tensor for medical image fusion," *Mul- tidimensional Systems and Signal Processing,* vol. 33, no. 3, p. 1001–1021, 2022. |
| [35] | Z. F. M. U. L. S. H. &. A. D. Ullah, "A hybrid image enhancement based brain MRI images classification technique," *Medical Hypotheses,* no. 143, p. 109922, 2020. |
| [36] | M. M. K. &. S. B. Veluchamy, "Brightness preserving optimized weighted bi-histogram equalization algorithm and its application to MR brain image segmentation," *International Journal of Imaging Systems and Technology,* vol. 29, no. 3, p. 339–352, 2019. |
| [37] | . Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," *Sixth International Conference on Computer Vision IEEE Cat. No.98CH36271,* vol. IEEE Cat. No.98CH36271, pp. 07-07, January 1998. |
| [38] | J. F. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vols. PAMI-8, no. 6, pp. 679 - 698, November 1986. |
| [39] | Z. Z. L. W. Weiguo Zhao, "Manta ray foraging optimization: An effective bio-inspired optimizer for engineering applications," *Engineering Applications of Artificial Intelligence,* 2019. |
| [40] | A. L. a. S. M. M. Seyedali Mirjalili, "The Whale Optimization Algorithm," 2016. |
| [41] | S. Mirjalili, "The Ant Lion Optimizer," 2015. |
| [42] | Z. L. B. L. K. &. L. F. Lu, "Effective guided image filtering for contrast enhancement," *IEEE Signal Processing Letters,* vol. 25, no. 10, p. 1585–1589 , 2018. |

1. A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, “Marine Predators Algorithm: A nature-inspired metaheuristic”, Expert Syst. Appl., vol. 152, 2020, doi: 10.1016/j.eswa.2020.113377. [↑](#footnote-ref-1)
2. Zuiderveld, K. (1994). Contrast limited adaptive histogram equalization. Graphics gems, 474-485. [↑](#footnote-ref-2)
3. https://drive.google.com/file/d/157bjO1\_cFuSd0HWDUuAmcHRJDVyWpOxB/view [↑](#footnote-ref-3)
4. https://drive.google.com/file/d/157bjO1\_cFuSd0HWDUuAmcHRJDVyWpOxB/view [↑](#footnote-ref-4)