**RELATED WORK**

**3. Chỉnh sửa ảnh Poisson hiệu quả bằng Pyramid**

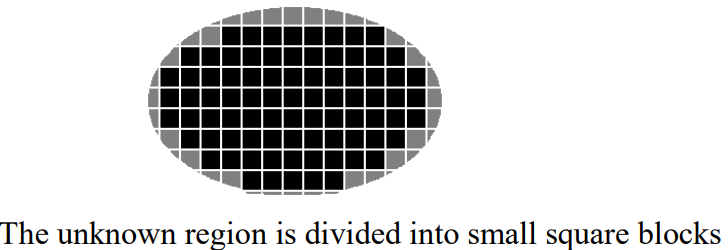
Việc tạo thành hai ảnh dựa trên phương trình Poisson yêu cầu giải một hệ tuyến tính với n ẩn số để tính giá trị của n pixel. Trong công trình này, một phương pháp thành phần ảnh được cải tiến dựa trên phương trình Poisson được giới thiệu để giảm thời gian thực hiện trong quá trình ghép ảnh. Các phương pháp đề xuất giải phương trình Poisson bằng cách sử dụng kim tự tháp hình ảnh và các phương pháp chia để trị. Kim tự tháp hình ảnh và các phương pháp chia để trị được sử dụng để tăng tốc thực hiện nhiều loại hình ảnh khác nhau.

Sử dụng các phương pháp đã đề xuất, thời gian để giải quyết một vấn đề lớn tại cùng một thời điểm thì lớn hơn so với thời gian thực hiện nếu nó được chia thành các vấn đề nhỏ hơn. Trong các phương pháp đề xuất, hai và ba cấp tháp được xây dựng để tăng hiệu quả của quá trình. Các phương pháp đề xuất bắt đầu từ cấp độ kim tự tháp thứ ba. Ở cấp thứ ba của kim tự tháp, hình ảnh nguồn và hình ảnh đích được kết hợp với nhau để tạo ra hình ảnh tổng hợp. Phương pháp PIE được sử dụng để tạo các ảnh nguồn và ảnh đích trong cấp độ thứ ba của kim tự tháp.4 Phương pháp này lấy các giá trị cường độ của các pixel ở viền thành các giá trị của hình ảnh đích. Sau đó, phương trình Poisson được giải để tính toán các giá trị cường độ cho các pixel trong vùng bên trong. Do đó, đường viền của đối tượng trong ảnh tổng hợp rất mịn. Sau khi tạo ra hình ảnh tổng hợp từ cấp kim tự tháp thứ ba, nó được sử dụng trong cấp thứ hai của kim tự tháp. Ở cấp độ thứ hai, ảnh nguồn và ảnh đích được kết hợp bằng cách giải phương trình Poisson lấy điều kiện biên Dirichlet từ kết quả ảnh tổng hợp từ cấp độ thứ ba. Vì vậy, hình ảnh tổng hợp được tạo ra từ cấp độ thứ hai là chân thực và mượt mà hơn nhiều. Ở cấp độ thứ hai, vùng chưa biết được chia thành một số vùng nhỏ chưa biết và hệ thống tuyến tính thưa thớt được giải cho từng vùng nhỏ riêng biệt. Cuối cùng, quá trình xảy ra ở cấp độ thứ hai được lặp lại ở cấp độ đầu tiên để tạo ra hình ảnh tổng hợp cuối cùng.

Như đã giải thích trước đây, các phương pháp được đề xuất dựa trên việc phân chia vùng chưa biết thành một số vùng nhỏ giải quyết hệ thống tuyến tính thưa thớt cho từng vùng nhỏ riêng biệt. Đối với vùng chưa biết, độ phức tạp của phương pháp thừa số hóa Lower-Upper (LU) để giải hệ tuyến tính trong phương trình là xấp xỉ trong đó kích thước của . Nếu vùng chưa biết được chia thành các vùng nhỏ, thì chi phí thời gian để giải hệ trong phương trình cho vùng chưa biết bằng cách sử dụng thừa số hóa LU bằng tổng chi phí thời gian để giải nó cho từng vùng nhỏ riêng biệt, như được minh họa trong phương trình sau:

trong đó là số pixel chưa biết trong vùng là số vùng nhỏ. Độ phức tạp để giải hệ thống tuyến tính cho tất cả các vùng một lần được cho bởi phương trình sau:

**3.2 Partitioning the unknown Region into Small Square Blocks (PRSSB)**

Trong phương pháp này, vùng chưa biết được chia thành các khối vuông nhỏ. Hình dạng của các vùng trong ranh giới của vùng không xác định là hình vuông không hoàn toàn (không phải tất cả các pixel trong khối đều là ẩn số) và hình dạng của các vùng bên trong là hình vuông (tất cả các pixel là ẩn số), như trong hình sau.

Sau đó , các Phương trình Poisson được giải cho từng vùng riêng biệt, lấy điều kiện biên Dirichlet từ hình ảnh đầu ra của mức kim tự tháp trước đó. Các bước được mô tả trong thuật toán sau:

*Thuật toán PRSSB: PIE hiệu quả dựa trên việc phân vùng không xác định thành các khối vuông nhỏ*

Inputs: mặt nạ (mask) của vùng chưa biết, hình ảnh nguồn và đích.

Output: hình ảnh tổng hợp cuối cùng.

1. Xây dựng các mức kim tự tháp cho các hình ảnh nguồn, đích và mặt nạ bằng cách sử dụng phương pháp kim tự tháp Gaussian.
2. Soạn ảnh nguồn và ảnh đích ở mức thứ ba của hình chóp bằng phương pháp PIE.
3. Ở cấp hai của hình chóp ta làm như sau:
   1. Chia mặt nạ của vùng chưa biết thành các khối vuông nhỏ, mỗi khối có kích thước .
   2. Tạo ma trận thưa thớt sẽ được sử dụng nếu tất cả các pixel trong khối là ẩn số bằng cách thực hiện các bước sau:
4. Khởi tạo ma trận với các phần tử size () và đường chéo 4s, trong đó là số pixel trong khối.
5. for c=0 to BS-1 do

for c=0 to BS-1 do

Xác định tập hợp NB các pixel lân cận xung quanh pixel P(r,c)

For px in NB1 do

if NB1 là pixel trên biên

SA(r,c)=0

else

SA(r,c)=-1

* 1. for block in mask do
     1. Soạn ảnh nguồn và ảnh đích bằng phương pháp PIE
     2. Giải phương trình Poisson với điều kiện biên Dirichlet lấy từ ảnh tổng hợp được tạo từ bước 1.
     3. if tất cả các điểm ảnh trên khối là ẩn số

Sử dụng ma trận SA được tính từ trước để giải hệ thống tuyến tính.

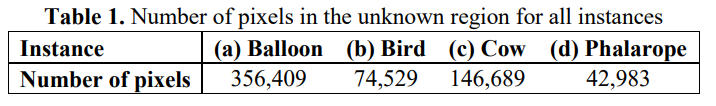
else

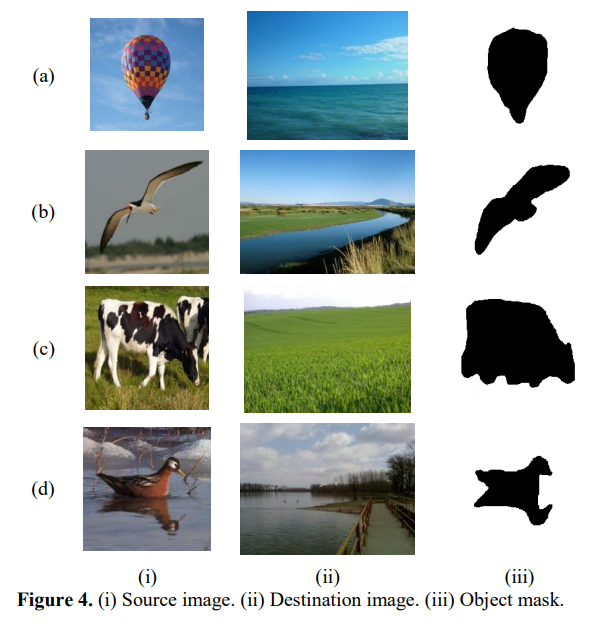
Tính ma trận thưa A để giải hệ thống tuyến tính.

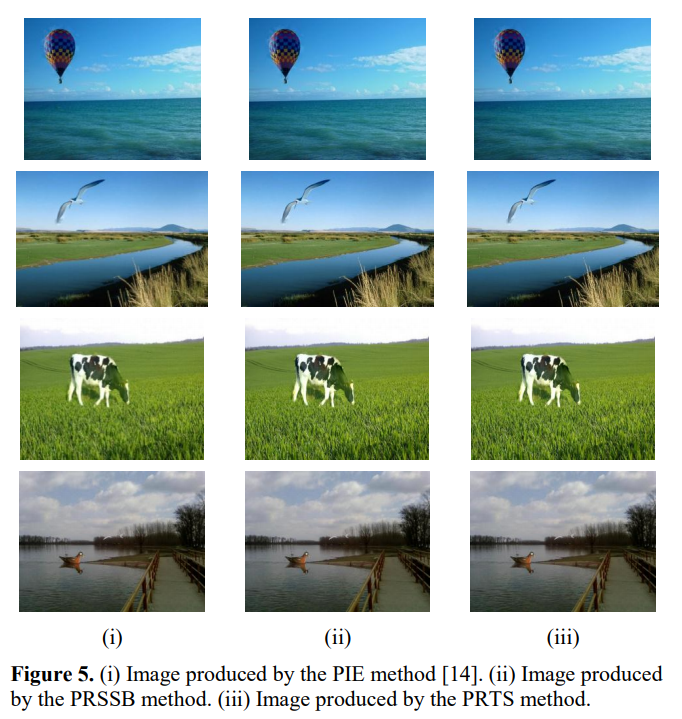
1. Tính toán hình ảnh tổng hợp cuối cùng bằng cách lặp lại bước trước đó ở cấp đầu tiên của hình chop

Kết quả tổng hợp

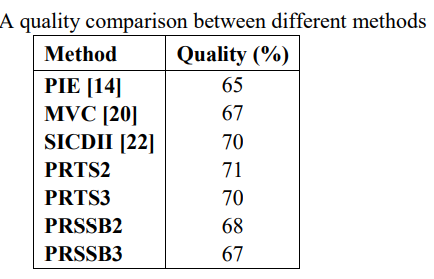
Bộ dữ liệu thực nghiệm:





Đây là kết quả thử nghiệm bộ dữ liệu trên bằng phương pháp truyền thống và 2 phương pháp cải tiến:

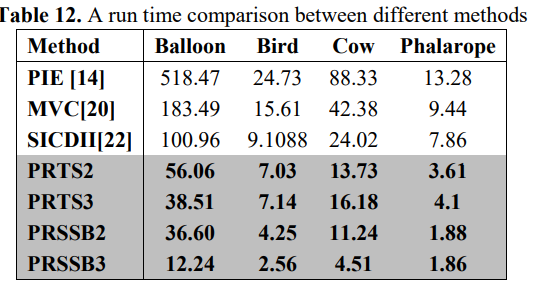
**Bảng thống kê kết quả thực nghiệm trên nhiều phương pháp khác nhau:**



Chất lượng giữa các phương pháp không khác nhau nhiều tức sự cải tiến này không cải tiến được quá nhiều về mặt hiếu suất:

PRSSB và PRTS chỉ hơn PIE truyền thống 2-3%. Tuy nhỏ nhưng cũng cho thấy phần nào sự tiến bộ trong thuật toán nhằm tang độ chính xác.

Thời gian mới là yếu tố chính được chú trọng của 2 phương pháp này:



Ta dễ dàng nhận ra rằng PRSSB3 nhỏ hơn PIE ở tập Balloon tận 43 lần. Hoặc số lần giảm ít nhất là 3.4 lần. Sự rút ngắn thời gian vượt ngoài mong đợi.

Qua đó cho ta thấy PRTS và PRSSB không làm tăng quá nhiều về mặt hiệu suất nhưng nó mang tính đột phá về thời gian chạy của thuật toán.