



# *ISP-DEMOSAICKING (中)*



食鱼者



202203



# 目录 CONTENTS

01. 算法原理精讲

02. 算法代码实现



*01*

# 算法原理精讲





# 算法精讲-方向加权法

- RB缺失的G通道的插值 1. 2. 3. 4 方向上  $k_n = 1$   
5-12 方向  $k_n = 0.5$  1/√5

Row

|                 |                 |                 |                 |                 |                 |                 |                 |                 |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| B <sub>00</sub> | G <sub>01</sub> | B <sub>02</sub> | G <sub>03</sub> | B <sub>04</sub> | G <sub>05</sub> | B <sub>06</sub> | G <sub>07</sub> | B <sub>08</sub> |
| G <sub>10</sub> | R <sub>11</sub> | G <sub>12</sub> | R <sub>13</sub> | G <sub>14</sub> | R <sub>15</sub> | G <sub>16</sub> | R <sub>17</sub> | G <sub>18</sub> |
| B <sub>20</sub> | G <sub>21</sub> | B <sub>22</sub> | G <sub>23</sub> | B <sub>24</sub> | G <sub>25</sub> | B <sub>26</sub> | G <sub>27</sub> | B <sub>28</sub> |
| G <sub>30</sub> | R <sub>31</sub> | G <sub>32</sub> | R <sub>33</sub> | G <sub>34</sub> | R <sub>35</sub> | G <sub>36</sub> | R <sub>37</sub> | G <sub>38</sub> |
| B <sub>40</sub> | G <sub>41</sub> | B <sub>42</sub> | G <sub>43</sub> | B <sub>44</sub> | G <sub>45</sub> | B <sub>46</sub> | G <sub>47</sub> | B <sub>48</sub> |
| G <sub>50</sub> | R <sub>51</sub> | G <sub>52</sub> | R <sub>53</sub> | G <sub>54</sub> | R <sub>55</sub> | G <sub>56</sub> | R <sub>57</sub> | G <sub>58</sub> |
| B <sub>60</sub> | G <sub>61</sub> | B <sub>62</sub> | G <sub>63</sub> | B <sub>64</sub> | G <sub>65</sub> | B <sub>66</sub> | G <sub>67</sub> | B <sub>68</sub> |
| G <sub>70</sub> | R <sub>71</sub> | G <sub>72</sub> | R <sub>73</sub> | G <sub>74</sub> | R <sub>75</sub> | G <sub>76</sub> | R <sub>77</sub> | G <sub>78</sub> |
| B <sub>80</sub> | G <sub>81</sub> | B <sub>82</sub> | G <sub>83</sub> | B <sub>84</sub> | G <sub>85</sub> | B <sub>86</sub> | G <sub>87</sub> | B <sub>88</sub> |

*(Note: The image shows a grid with handwritten annotations. A blue line connects (0,0) to (4,4). A red line connects (4,4) to (8,8). Numbers 1-12 are written around the center pixel (4,4) to indicate directions. A blue circle highlights the center pixel (4,4).)*

| <i>n</i> | <i>v<sub>n</sub></i> | <i>h<sub>n</sub></i> | <i>n</i> | <i>v<sub>n</sub></i> | <i>h<sub>n</sub></i> |
|----------|----------------------|----------------------|----------|----------------------|----------------------|
| 1        | 0                    | -1                   | 2        | -1                   | 0                    |
| 3        | 0                    | +1                   | 4        | +1                   | 0                    |
| 5        | -1                   | -2                   | 6        | -2                   | -1                   |
| 7        | -2                   | +1                   | 8        | -1                   | +2                   |
| 9        | +1                   | +2                   | 10       | +2                   | +1                   |
| 11       | +2                   | -1                   | 12       | +1                   | -2                   |

$$In(i,j) = k_n \{ \text{abs}[P(i+v_n, j+h_n) - P(i-v_n, j-h_n)] + \text{abs}[P(i+2v_n, j+2h_n) - P(i,j)] \}$$

$$k_n (|P(4,3) - P(4,5)| + |P(4,2) - P(4,4)|)$$

$$Is(4,4) = k_n (|P(3,2) - P(5,6)| + |P(2,0) - P(4,4)|)$$

12个方向  $In$   $k_n = 0.5$   
 类梯度值

$$Wn(i,j) = \left( \frac{1}{1 + In(i,j)} \right) \left( \sum \left( \frac{1}{1 + In(i,j)} \right) \right)^{-1}$$



# 算法精讲-方向加权法

- RB缺失的G通道的插值

|                 |                 |                 |                 |                 |                 |                 |                 |                 |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| B <sub>00</sub> | G <sub>01</sub> | B <sub>02</sub> | G <sub>03</sub> | B <sub>04</sub> | G <sub>05</sub> | B <sub>06</sub> | G <sub>07</sub> | B <sub>08</sub> |
| G <sub>10</sub> | R <sub>11</sub> | G <sub>12</sub> | R <sub>13</sub> | G <sub>14</sub> | R <sub>15</sub> | G <sub>16</sub> | R <sub>17</sub> | G <sub>18</sub> |
| B <sub>20</sub> | G <sub>21</sub> | B <sub>22</sub> | G <sub>23</sub> | B <sub>24</sub> | G <sub>25</sub> | B <sub>26</sub> | G <sub>27</sub> | B <sub>28</sub> |
| G <sub>30</sub> | R <sub>31</sub> | G <sub>32</sub> | R <sub>33</sub> | G <sub>34</sub> | R <sub>35</sub> | G <sub>36</sub> | R <sub>37</sub> | G <sub>38</sub> |
| B <sub>40</sub> | G <sub>41</sub> | B <sub>42</sub> | G <sub>43</sub> | B <sub>44</sub> | G <sub>45</sub> | B <sub>46</sub> | G <sub>47</sub> | B <sub>48</sub> |
| G <sub>50</sub> | R <sub>51</sub> | G <sub>52</sub> | R <sub>53</sub> | G <sub>54</sub> | R <sub>55</sub> | G <sub>56</sub> | R <sub>57</sub> | G <sub>58</sub> |
| B <sub>60</sub> | G <sub>61</sub> | B <sub>62</sub> | G <sub>63</sub> | B <sub>64</sub> | G <sub>65</sub> | B <sub>66</sub> | G <sub>67</sub> | B <sub>68</sub> |
| G <sub>70</sub> | R <sub>71</sub> | G <sub>72</sub> | R <sub>73</sub> | G <sub>74</sub> | R <sub>75</sub> | G <sub>76</sub> | R <sub>77</sub> | G <sub>78</sub> |
| B <sub>80</sub> | G <sub>81</sub> | B <sub>82</sub> | G <sub>83</sub> | B <sub>84</sub> | G <sub>85</sub> | B <sub>86</sub> | G <sub>87</sub> | B <sub>88</sub> |

B上的G的插值

R →

| n  | v <sub>n</sub> | h <sub>n</sub> | n  | v <sub>n</sub> | h <sub>n</sub> |
|----|----------------|----------------|----|----------------|----------------|
| 1  | 0              | -1             | 2  | -1             | 0              |
| 3  | 0              | +1             | 4  | +1             | 0              |
| 5  | -1             | -2             | 6  | -2             | -1             |
| 7  | -2             | +1             | 8  | -1             | +2             |
| 9  | +1             | +2             | 10 | +2             | +1             |
| 11 | +2             | -1             | 12 | +1             | -2             |

W<sub>n</sub>(i,j)

$$G(i,j) = B(i,j) + \sum_{n=1}^{12} W_n(i,j) * K_{b,n}(i+v_n, j+h_n)$$

$G(i,j) - B(i,j) = \Delta \text{色差}$

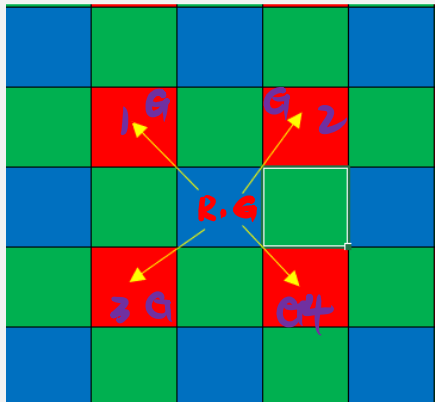
$$K_{b,n}(i+v_n, j+h_n) = G(i+v_n, j+h_n) - B(i+v_n, j+h_n)$$

左右或上下的B的mean



# 算法精讲-方向加权法

- RB缺失的BR的插值



~~B~~ R  $\Rightarrow$  B (Row)

TABLE 2. POSITIONS OF NEARBY SAMPLES IN STEP 2.

| $n$ | $v'_n$ | $h'_n$ | $n$ | $v'_n$ | $h'_n$ |
|-----|--------|--------|-----|--------|--------|
| 1   | -1     | -1     | 2   | -1     | +1     |
| 3   | +1     | +1     | 4   | +1     | -1     |

$$Inc(i, j) = abs[p(i + v'_n, j + h'_n) - p(i - v'_n, j - h'_n)] + abs[p(i + 2v'_n, j + 2h'_n) - p(i, j)]$$

$$W_n(i, j) = \frac{1}{1 + Inc(i, j)} / \sum_{n=1}^4 \frac{1}{1 + Inc(i, j)}$$

$$R(i, j) = G(i, j) - \sum_{n=1}^4 W_n(i, j) \cdot K_n(i + v'_n, j + h'_n)$$

$$K_n(i + v'_n, j + h'_n) = \frac{G(i + v'_n, j + h'_n) - R(i + v'_n, j + h'_n)}{\Delta}$$



# 算法精讲-方向加权法

- G缺失的BR的插值

| $n$ | $v_n$ | $h_n$ | $n$ | $v_n$ | $h_n$ |
|-----|-------|-------|-----|-------|-------|
| 1   | 0     | -1    | 2   | -1    | 0     |
| 3   | 0     | +1    | 4   | +1    | 0     |
| 5   | -1    | -2    | 6   | -2    | -1    |
| 7   | -2    | +1    | 8   | -1    | +2    |
| 9   | +1    | +2    | 10  | +2    | +1    |
| 11  | +2    | -1    | 12  | +1    | -2    |



$$I_n(i, j) = \frac{\kappa_n(\text{abs}(P(i + v_n, j + h_n) - P(i - v_n, j - h_n))) + \text{abs}(P(i + 2v_n, j + 2h_n) - P(i, j)))}{\text{abs}(P(i + 2v_n, j + 2h_n) - P(i, j))}$$

$$w_n(i, j) = \left( \frac{1}{1 + I_n(i, j)} \right) / \sum_{n=1}^{12} \frac{1}{1 + I_n(i, j)},$$

$$R(i, j) = G(i, j) - \sum_{n=1}^{12} w_n(i, j) * K_{r,n}(i + v_n, j + h_n),$$

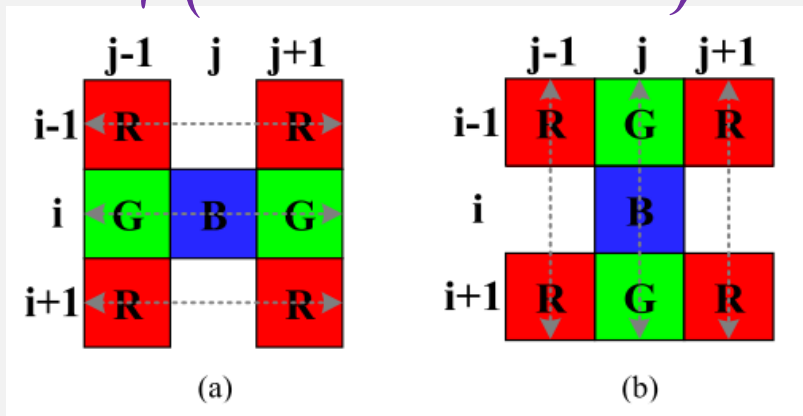


## 算法精讲-基于边缘检测的方法

- 计算水平和垂直梯度

$$\begin{cases} e_h = \frac{1}{4} (|C_{i,j} - C_{i-1,j}| + |C_{i,j} - C_{i+1,j}| + |C_{i,j} - C_{i,j-1}| + |C_{i,j} - C_{i,j+1}|) \\ e_v = \frac{1}{4} (|C_{i,j} - C_{i,j-1}| + |C_{i,j} - C_{i,j+1}| + |C_{i,j} - C_{i-1,j}| + |C_{i,j} - C_{i+1,j}|) \end{cases}$$

|           |           |           |           |           |           |
|-----------|-----------|-----------|-----------|-----------|-----------|
| $B_{0,0}$ | $G_{0,1}$ | $B_{0,2}$ | $G_{0,3}$ | $B_{0,4}$ | $G_{0,5}$ |
| $G_{1,0}$ | $R_{1,1}$ | $G_{1,2}$ | $R_{1,3}$ | $G_{1,4}$ | $R_{1,5}$ |
| $B_{2,0}$ | $G_{2,1}$ | $B_{2,2}$ | $G_{2,3}$ | $B_{2,4}$ | $G_{2,5}$ |
| $G_{3,0}$ | $R_{3,1}$ | $G_{3,2}$ | $R_{3,3}$ | $G_{3,4}$ | $R_{3,5}$ |
| $B_{4,0}$ | $G_{4,1}$ | $B_{4,2}$ | $G_{4,3}$ | $B_{4,4}$ | $G_{4,5}$ |
| $G_{5,0}$ | $R_{5,1}$ | $G_{5,2}$ | $R_{5,3}$ | $G_{5,4}$ | $R_{5,5}$ |







## 算法精讲-基于边缘检测的方法

- 计算缺失的G

$$W_h = \frac{\frac{1}{e_h}}{\frac{1}{e_h} + \frac{1}{e_v}} \quad W_v = \frac{\frac{1}{e_v}}{\frac{1}{e_h} + \frac{1}{e_v}}$$

|           |           |           |           |           |           |
|-----------|-----------|-----------|-----------|-----------|-----------|
| $B_{0,0}$ | $G_{0,1}$ | $B_{0,2}$ | $G_{0,3}$ | $B_{0,4}$ | $G_{0,5}$ |
| $G_{1,0}$ | $R_{1,1}$ | $G_{1,2}$ | $R_{1,3}$ | $G_{1,4}$ | $R_{1,5}$ |
| $B_{2,0}$ | $G_{2,1}$ | $B_{2,2}$ | $G_{2,3}$ | $B_{2,4}$ | $G_{2,5}$ |
| $G_{3,0}$ | $R_{3,1}$ | $G_{3,2}$ | $R_{3,3}$ | $G_{3,4}$ | $R_{3,5}$ |
| $B_{4,0}$ | $G_{4,1}$ | $B_{4,2}$ | $G_{4,3}$ | $B_{4,4}$ | $G_{4,5}$ |
| $G_{5,0}$ | $R_{5,1}$ | $G_{5,2}$ | $R_{5,3}$ | $G_{5,4}$ | $R_{5,5}$ |



$$\begin{cases} \hat{G}_h(i, j) = \tilde{G}_h(i, j) + \Delta C_h(i, j) \\ \hat{G}_v(i, j) = \tilde{G}_v(i, j) + \Delta C_v(i, j) \end{cases}$$

$$\Delta G_h(i, j) = \frac{1}{2} \left( \frac{1}{2} (C_{ii,j} - C_{ii,j+2}) + \right.$$

$$\left. \frac{1}{2} (C_{ii,j} - C_{ii,j+2}) \right)$$

$$G'(i, j) = W_h(i, j) * \hat{G}_h(i, j) + W_v(i, j) * \hat{G}_v(i, j)$$



# 算法精讲-基于边缘检测的方法

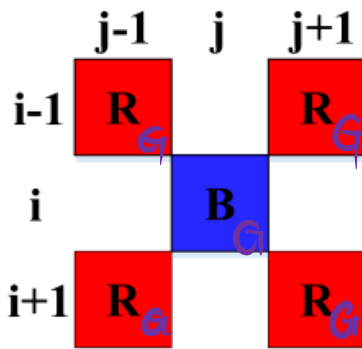
- 计算缺失的RB

$R \Rightarrow B$

$B \Rightarrow R$

|           |           |           |           |           |           |
|-----------|-----------|-----------|-----------|-----------|-----------|
| $B_{0,0}$ | $G_{0,1}$ | $B_{0,2}$ | $G_{0,3}$ | $B_{0,4}$ | $G_{0,5}$ |
| $G_{1,0}$ | $R_{1,1}$ | $G_{1,2}$ | $R_{1,3}$ | $G_{1,4}$ | $R_{1,5}$ |
| $B_{2,0}$ | $G_{2,1}$ | $B_{2,2}$ | $G_{2,3}$ | $B_{2,4}$ | $G_{2,5}$ |
| $G_{3,0}$ | $R_{3,1}$ | $G_{3,2}$ | $R_{3,3}$ | $G_{3,4}$ | $R_{3,5}$ |
| $B_{4,0}$ | $G_{4,1}$ | $B_{4,2}$ | $G_{4,3}$ | $B_{4,4}$ | $G_{4,5}$ |
| $G_{5,0}$ | $R_{5,1}$ | $G_{5,2}$ | $R_{5,3}$ | $G_{5,4}$ | $R_{5,5}$ |

$G \rightarrow BR$



$$d_{GC} = \frac{1}{n} \sum G'(i+k, j+l) - C(i+k, j+l)$$

$$R'(i, j) = G'(i, j) + d_{GC}$$



## 算法精讲-自适应边缘增强法

- 计算缺失的G

|           |           |           |           |           |           |
|-----------|-----------|-----------|-----------|-----------|-----------|
| $B_{0,0}$ | $G_{0,1}$ | $B_{0,2}$ | $G_{0,3}$ | $B_{0,4}$ | $G_{0,5}$ |
| $G_{1,0}$ | $R_{1,1}$ | $G_{1,2}$ | $R_{1,3}$ | $G_{1,4}$ | $R_{1,5}$ |
| $B_{2,0}$ | $G_{2,1}$ | $B_{2,2}$ | $G_{2,3}$ | $B_{2,4}$ | $G_{2,5}$ |
| $G_{3,0}$ | $R_{3,1}$ | $G_{3,2}$ | $R_{3,3}$ | $G_{3,4}$ | $R_{3,5}$ |
| $B_{4,0}$ | $G_{4,1}$ | $B_{4,2}$ | $G_{4,3}$ | $B_{4,4}$ | $G_{4,5}$ |
| $G_{5,0}$ | $R_{5,1}$ | $G_{5,2}$ | $R_{5,3}$ | $G_{5,4}$ | $R_{5,5}$ |

$$DV_{i,j} = |RB(i,j-1) - RB(i+1,j-1)| \\ + |G(i+1,j) - G(i+1,j+1)| + |RB(i+1,j+1) - RB(i+1,j+2)|$$

$$DH(i,j)$$

$$TD(i,j) = DV + DH$$



# 算法精讲-自适应边缘增强法

- 计算缺失的G

|           |           |           |           |           |           |
|-----------|-----------|-----------|-----------|-----------|-----------|
| $B_{0,0}$ | $G_{0,1}$ | $B_{0,2}$ | $G_{0,3}$ | $B_{0,4}$ | $G_{0,5}$ |
| $G_{1,0}$ | $R_{1,1}$ | $G_{1,2}$ | $R_{1,3}$ | $G_{1,4}$ | $R_{1,5}$ |
| $B_{2,0}$ | $G_{2,1}$ | $B_{2,2}$ | $G_{2,3}$ | $B_{2,4}$ | $G_{2,5}$ |
| $G_{3,0}$ | $R_{3,1}$ | $G_{3,2}$ | $R_{3,3}$ | $G_{3,4}$ | $R_{3,5}$ |
| $B_{4,0}$ | $G_{4,1}$ | $B_{4,2}$ | $G_{4,3}$ | $B_{4,4}$ | $G_{4,5}$ |
| $G_{5,0}$ | $R_{5,1}$ | $G_{5,2}$ | $R_{5,3}$ | $G_{5,4}$ | $R_{5,5}$ |

$$TD = DH + DV$$

$TD < Th$  那么平坦

$$G(i,j) = \frac{3}{8} (G(i,j-1) + G(i,j+1)) + \frac{1}{8} (G(i-1,j) + G(i+1,j)) + R - \frac{1}{2} (\frac{1}{2} (R + R(i,j-2)) + \frac{1}{2} (R + R(i,j+2)))$$

$$TD > Th, DH < DV$$

$$G(i,j) = \frac{1}{2} (G(i,j-1) + G(i,j+1)) + (R - \bar{R})$$

$$TD > Th, DH < DV$$

$$G(i,j) = \frac{1}{8} \bar{G}_H + \frac{3}{8} \bar{G}_V + \frac{1}{2} (R - \bar{R})$$



# 算法精讲-自适应边缘增强法

- 计算缺失的RB

|           |           |           |           |           |           |
|-----------|-----------|-----------|-----------|-----------|-----------|
| $B_{0,0}$ | $G_{0,1}$ | $B_{0,2}$ | $G_{0,3}$ | $B_{0,4}$ | $G_{0,5}$ |
| $G_{1,0}$ | $R_{1,1}$ | $G_{1,2}$ | $R_{1,3}$ | $G_{1,4}$ | $R_{1,5}$ |
| $B_{2,0}$ | $G_{2,1}$ | $B_{2,2}$ | $G_{2,3}$ | $B_{2,4}$ | $G_{2,5}$ |
| $G_{3,0}$ | $R_{3,1}$ | $G_{3,2}$ | $R_{3,3}$ | $G_{3,4}$ | $R_{3,5}$ |
| $B_{4,0}$ | $G_{4,1}$ | $B_{4,2}$ | $G_{4,3}$ | $B_{4,4}$ | $G_{4,5}$ |
| $G_{5,0}$ | $R_{5,1}$ | $G_{5,2}$ | $R_{5,3}$ | $G_{5,4}$ | $R_{5,5}$ |

$TD < Th$  平坦区

迭代

$$\underline{B(3,3) = \bar{B}_n + G_{(3,3)} - \bar{G}_n}$$

$DH < DV$

$$B(3,3) = \bar{B}_n + G(3,3) - \frac{1}{8} ( 3[G(i-1,j) + G(i+1,j) + G(i,j-1) + G(i,j+1)] )$$

$DH > DV$

$$\bar{B}_n + G(3,3) - \frac{1}{8} ( G(i-1,j) + G(i+1,j) + 3(G(i,j+1) + G(i,j-1)) )$$



# 算法精讲-自适应边缘增强法

- 计算缺失的RB

$$\frac{1111}{0111}$$

|           |           |           |           |           |           |
|-----------|-----------|-----------|-----------|-----------|-----------|
| $B_{0,0}$ | $G_{0,1}$ | $B_{0,2}$ | $G_{0,3}$ | $B_{0,4}$ | $G_{0,5}$ |
| $G_{1,0}$ | $R_{1,1}$ | $G_{1,2}$ | $R_{1,3}$ | $G_{1,4}$ | $R_{1,5}$ |
| $B_{2,0}$ | $G_{2,1}$ | $B_{2,2}$ | $G_{2,3}$ | $B_{2,4}$ | $G_{2,5}$ |
| $G_{3,0}$ | $R_{3,1}$ | $G_{3,2}$ | $R_{3,3}$ | $G_{3,4}$ | $R_{3,5}$ |
| $B_{4,0}$ | $G_{4,1}$ | $B_{4,2}$ | $G_{4,3}$ | $B_{4,4}$ | $G_{4,5}$ |
| $G_{5,0}$ | $R_{5,1}$ | $G_{5,2}$ | $R_{5,3}$ | $G_{5,4}$ | $R_{5,5}$ |

$$\frac{1}{2} \quad \frac{1}{4} \quad \frac{1}{8}$$

$$RB(i,j) = \frac{1}{2}(RB(i-1,j) + RB(i+1,j)) + \frac{1}{2}G(i,j) - \frac{1}{8}[(G(i-1,j-1) + G(i-1,j+1) + G(i+1,j-1) + G(i+1,j+1))]$$

$$\frac{1}{2}(RB(i,j+1) + RB(i,j-1)) + G(i,j) - \frac{1}{2}(G(i,j+1) + G(i,j-1))$$

移位 2倍 2<sup>n</sup>



02

## 算法代码实现





食鱼者



202203



wtzhu13



<https://gitee.com/wtzhu13>



猪猪爱吃鱼



wtzhu\_\_13

*See You!*



食鱼者



扫一扫上面的二维码图案，加我微信