УДК 621.376.56

ОБРОБКА КОЛЬОРОВИХ ЗОБРАЖЕНЬ НА ОСНОВІ СТВОРЕННЯ АДАПТИВНОЇ ПАЛІТРИ

© Таянов С.А., 2001
Івано-Франківський державний технічний університет нафти і газу

© Мельничук С.І, 2001
НУ "Львівська політехніка"

Представлена методика обробки та стиску кольорових зображень на основі створення адаптивної палітри. Пропонується алгоритм створення адаптивної палітри, наведений метод перетворення кольору пікселя зображення в RGB форматі в формат адаптивної палітри.

При створенні автоматизованих систем розпізнавання об'єктів існує проблема розпізнавання кольорових зображень. Системи штучного зору дають похибки при оцифровуванні кольорів. Тобто один і той самий колір може бути перетворений в різні (сусідні) кольори, так як стандартна 24 або 32 бітна палітра RGB включає в себе відповідно 2^{24} або 2^{32} кольорів та їх відтінків. У багатьох випадках така кількість кольорів є надлишковою і не відповідає дійсності. Зменшення кількості кольорів у кольоровій палітрі полегшить розпізнавання та подальшу обробку зображення, а також призведе до стиску цього зображення, так як для представлення однієї точки буде використано меншу кількість біт. Крім того, подальша архівація файлу зображення стандартними архіваторами (zip, arj, rar) призведе до подальшого стиску цього файлу за рахунок повторюваності даних.

Крім того використання для обробки RGB палітри не завжди є виправданим, так як RGB не дає чіткого розмежування по кольорах.

Тому розробка принципів побудови адаптивної палітри кольорів для конкретного зображення є актуальною.

Для початку розглянемо 24 бітну палітру RGB. Ця палітра найбільш розповсюджена, всі основні графічні формати її підтримують. Колір в цій палітрі задається 3 байтами: 1-й старший байт – синій колір (Blue), 2-й байт – зелений (Green) та 3-й байт – червоний (Red). Тут значення R, G та B змінюються від 0 до 255.

Розглянемо палітру HSB (інколи вона називається HSV). HSB (Hue, Saturation, Brightness) палітра дає можливість відокремлювати окремі кольори та їх відтінки. Тут H (колір) змінюється від 0 до 359, значення S (насиченість) від 0 до 1, значення B (яскравість) від 0 до 255.

Нижче (на рис. 1) приводиться блок-схема програми, яка використовувалася для перетворення кольорів R, G, і B в кольори H, S, і B для кожної точки зображення. Тут функції min() та max() шукають мінімальне та максимальне значення серед трьох змінних відповідно, B (Brightness) позначено літерою V.

Для візуального представлення результатів перетворення необхідне зворотне перетворення кольорів. Блок-схема алгоритму перетворення у представлена на рис. 2.

Тут функція ENT() виділяє цілу частину H.

Колір HSB палітри пропонується представляти у вигляді цілого числа, яке змінюється від 0 до 3599999. Тут молодші два десяткові розряди – це Brightness, яке змінюється від 0 до 99, далі два розряди Saturation – від 0 до 99 та три останніх розряди – це Hue – від 0 до 359. Для того, щоб представити колір в прийнятій HSB, палітрі необхідно перетворити кольори R, G та B в кольори H, S та B. Після цього необхідно масштабувати значення S та B так, щоб вони змінювалися в діапазоні від 0 до 99 та були цілими числами.

Тепер розглянемо алгоритм створення адаптивної палітри кольорів. Вхідними даними для визначення палітри буде кількість кольорів n, допуск (тут мається на увазі величина, яка обернена до роздільної здатності) по Hue – dH, по Saturation – dS та по Brightness – dB. Чим більшою є величина допуску, тим меншою є роздільна здатність по відповідній величині. Послідовність створення палітри наступна:

1) визначення залежності кількості точок зображення певного кольору від номеру кольору HSB. Запишемо цю залежність в масив і назвемо його NH (розмір масиву 3600000). Тоді NH[i] – це кількість точок і-го кольору прийнятої HSB-палітри;

2) створення безпосередньо адаптивної палітри для даного зображення, блок-схема алгоритму приведена на рис.3.

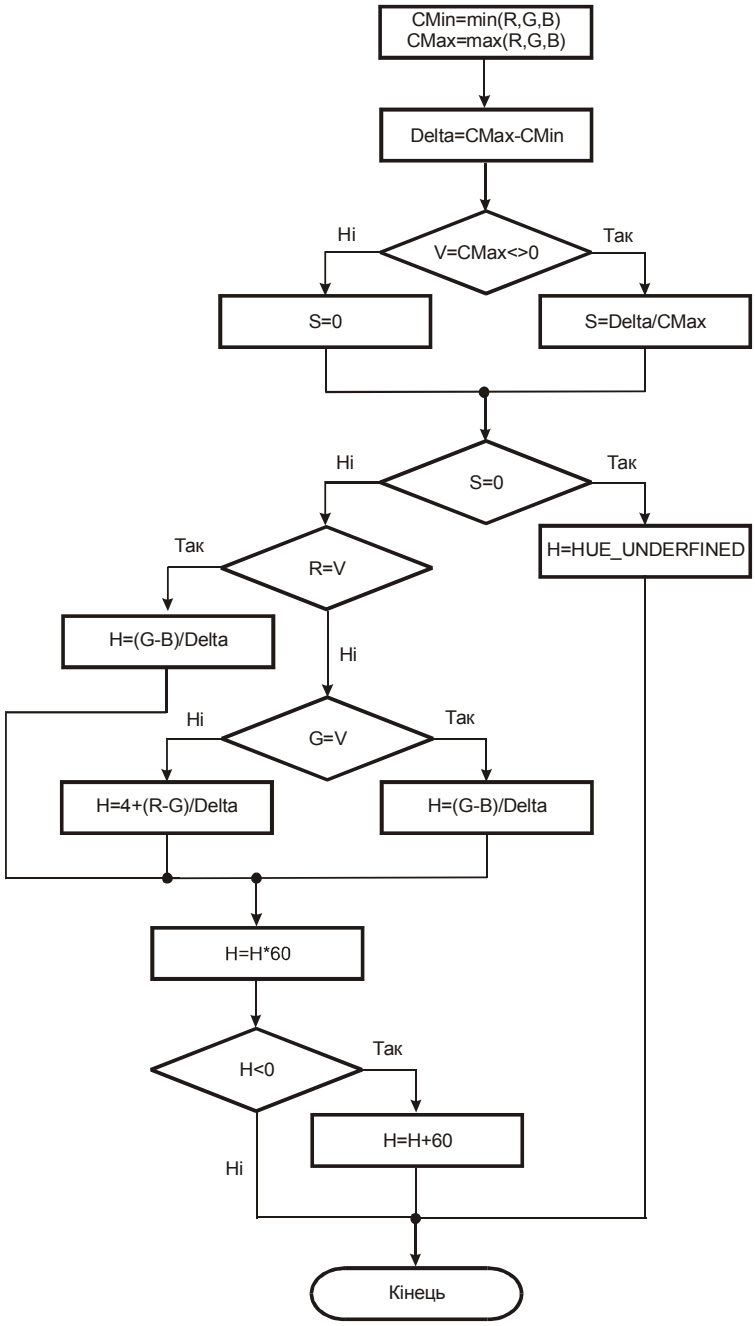

Рис. 1. Блок-схема перетворення кольорів RGB в HSB.

Тут AP[i] – і-й елемент масиву палітри, функція maxi() визначає номер елементу масиву NH, який має максимальне значення, n-кількість елементів палітри AP. Масив NH обнулюється в деякому околі від знайденого значення елементу масиву палітри так щоб запобігти використанню сусідніх кольорів в адаптивній палітрі стосовно S, H та B. Кількість елементів, що обнулюються, визначається допусками dH, dS та dB. Так як номер елементу масиву NH – це певний колір HSB, а величина і-го елементу масиву AP- це так само колір HSB, то умову обнулення можна записати наступним чином:

$$\begin{aligned}&(|H_{NH} - H1| < dH) \,\&\, (|S_{NH} - S1| < \\ &< dS) \,\&\, (|B_{NH} - B1| < dB)\end{aligned}, \qquad (1)$$

де H1, S1, B1 – кольори палітри HSB, що відповідають величині і-того елементу масиву AP; H_{NH}, S_{NH}, B_{NH}– кольори палітри HSB, що відповідають номеру елементу масиву NH.

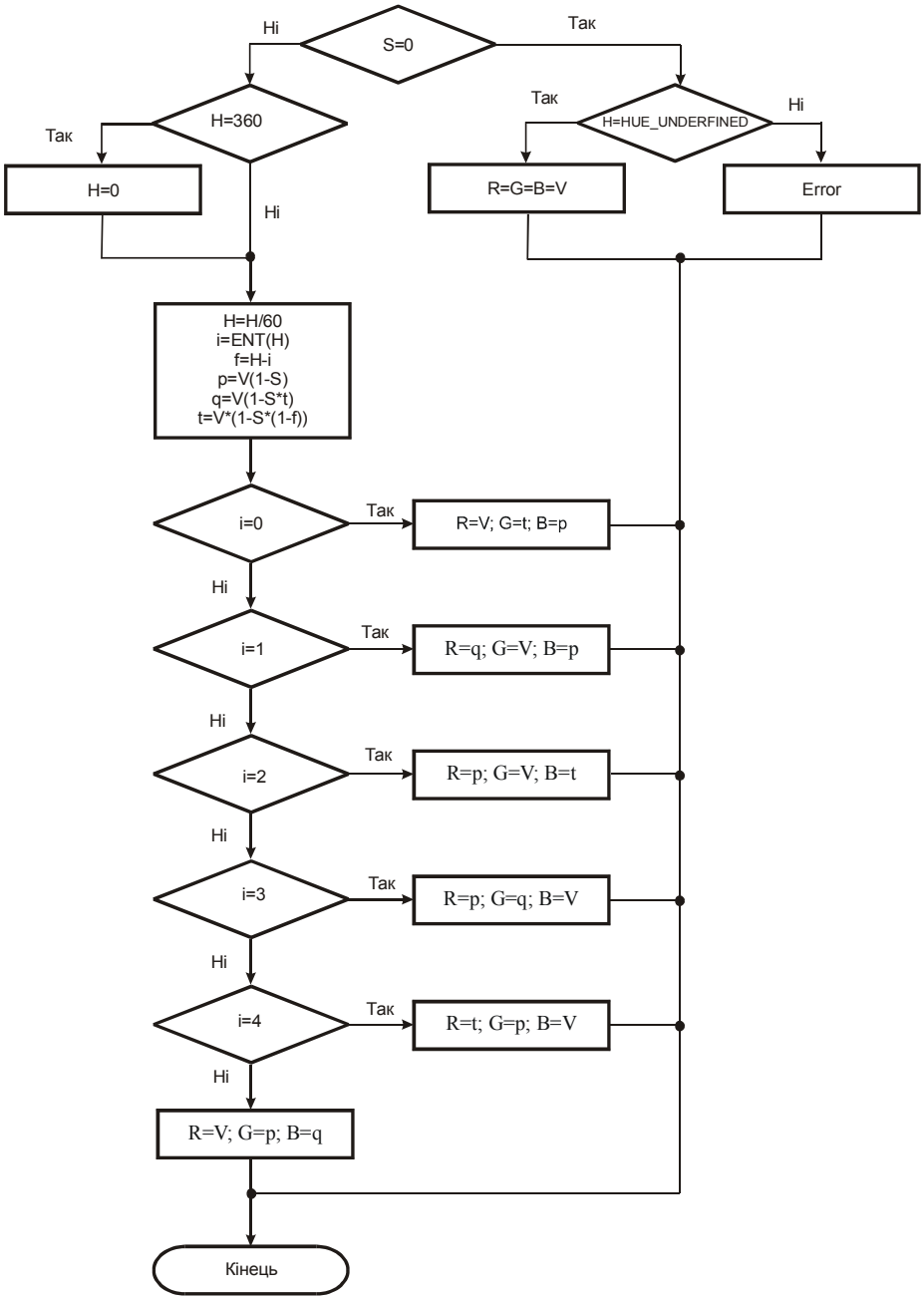

Рис. 2. Блок-схема перетворення кольорів HSB в RGB.

Якщо ця умова виконується, то відповідний елемент масиву NH обнулюється і його величина нехтується при подальшому визначення адаптивної палітри.

На рис. 4. графічно показано прямокутник ко-льорів палітри HSB, які не будуть враховуватись при визначенні наступних елементів адаптивної кольорової палітри AP.

Практично алгоритм виглядає наступним чи-ном:

```
for (i=1;i<=n; i++)
{
AP[i]=maxi(NH);
for(int j=AP[i]-dH*10000; j<=Hue[i]+dH; j+=10000)
      for(int j1=j-dS*100; j1=j+dS*100; j1+=100)
            for(int j2=j1-dV; j2<=j1+dV; j2++)
                  if(j2<3600000 && j2>0)
                        NH[j2]=0;
}
```

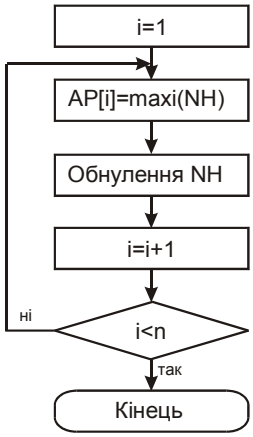

Рис.3. Блок-схема створення адаптивної палітри.

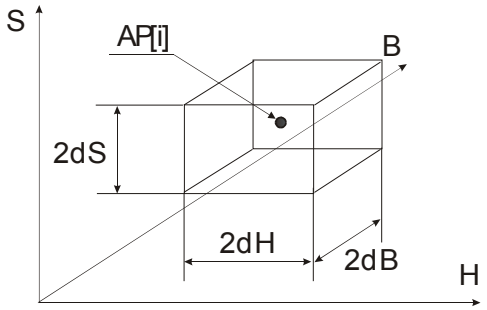

Рис. 4. Обнулення масиву NH.

3) перетворення кольору точки, який представлений в палітрі RGB, в колір адаптивної палітри. Блок-схема алгоритму наведена на рис. 5. Тут ColorHSB – номер кольору точки зображення за адаптивною палітрою AP, параметр *L* визначається наступним чином:

$$L = \sqrt{(H - H1)^2 + (S - S1)^2 + (B - B1)^2}\ , \qquad (2)$$

де *H, S, B* – колір певного пікселя зображення; *H1, B1, S1* – колір адаптивної палітри AP.

Суть алгоритму полягає в тому, що кожному пікселю зображення присвоюється колір адаптивної палітри, в якого значення *L* мінімальне для даного пікселя.

Зображення можна записати у вигляді масиву, де кожній точці зображення буде відповідати номер кольору адаптивної палітри;

4) необхідно враховувати, що при значенні B (Brightness) меншому від певної величини, колір точки можна розглядати як чорний. Тоді величини S (Saturation) та H (Hue) не мають значення. Можна прийняти, що H=0, S=0, B=0, тобто HSB=0. Крім того, якщо величина S деякої точки є менша певного значення, то колір можна відносити до чорно-білого. Тоді величина H немає значення і можна прийняти, що H=0 та S=0. В цьому випадку HSB буде змінюватися в діапазоні від 0 до 99. Вище вказані зауваження дають можливість усунути повторення чорного та сірих кольорів в палітрі AP.

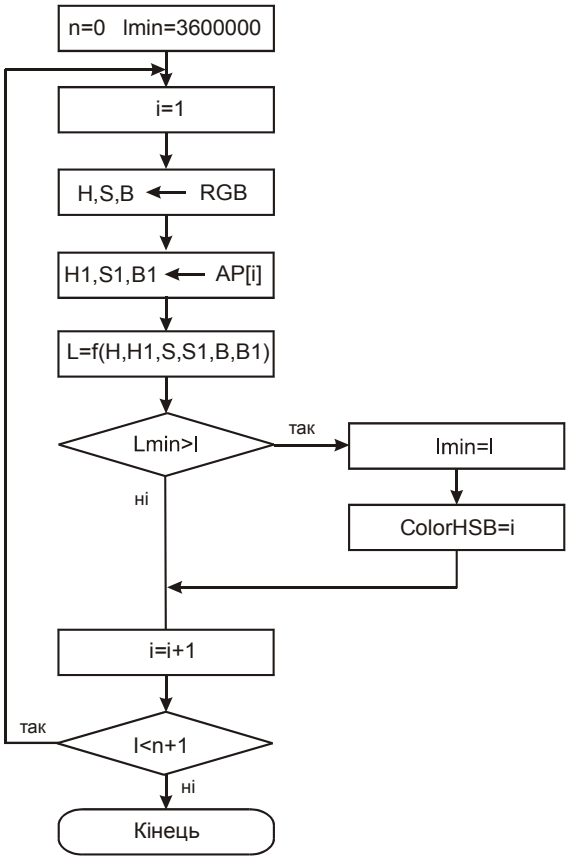

Рис.5. Блок-схема перетворення кольорів зображення в адаптивну палітру.

Дослідження показали, що можна зменшити кількість кольорів кольорового зображення з 20000-40000 до 100 без втрати інформації, яка необхідна для подальшого розпізнавання та обробки зображення. Це дасть можливість зменшити обсяг інформації, що передається, та спростити обробку зображення.

1. Гунтер Борн. Форматы данных: Пер. с нем. - К.: Торгово-издательское бюро BHV, 1995.-472с.