

Ю.П. КРИВЕНЧУК, С.В. ГЕЛЕТІЙ  
Національний університет "Львівська політехніка"

## КОНЦЕПЦІЯ ПЕРЕТВОРЕННЯ ТЕКСТУ В БІТОВУ КАРТУ З ВИКОРИСТАННЯМ БУДЬ-ЯКОГО ШРИФТУ

Сучасний світ неможливо уявити без обміну інформацією. Газети, журнали, книги, друкована рекламна продукція відіграють велику роль та увесь час удосконалюються та вимагають все більших ресурсів та можливостей. На даний момент, через різке зростання цифрових технологій виникла проблема між можливостями апаратного та програмного забезпечення. Тобто при передачі даних з електронного світу на паперовий відображається різниця між обчислювальними можливостями апаратних та програмних засобів. Це стосується засобів друку, які не можуть передати весь об'єм та глибину зображуваного на екрані. Для друку швидше використовують растрові шрифти, основною вадою яких є їхня немасштабованість. При дослідженні шрифтів у вбудованих системах виникли проблеми з тим, що існує досить невелика кількість готових варіантів перетворення тексту у бітову карту без певних обмежень. Шрифт, який виводиться на екрані та безпосередньо друкується принтером чи іншим пристроєм може суттєво відрізнятись, оскільки кількість пікселів, які можуть бути надрукованими є завеликою. Тому виникає необхідність оптимізації перетворення шрифту у бітову карту, тобто зменшення кількості точок без втрати основи гліфу. У роботі наведено перетворення шрифту у бітову карту за допомогою програми, у якій за основу було взято лінеатуру, тобто поділ зображення на лінії, який використовується у поліграфії. Для того, щоб створити необхідне растрове зображення у програмі потрібно вказати певний гліф та його розмір, а також пристрій, для якого він буде використовуватись. Перетворювач сканування приймає контур і застосовує набір правил, щоб визначити, які пікселі будуть частиною зображення гліфа при друку або відображенні на екрані. Отримані результати дозволяють друкувати вподобані нами шрифти, замість стандартних.

Ключові слова: шрифти, бітова карта, растрове зображення, лінеатура.

YU.P. KRYVENCHUK, S.V. HELETII  
Lviv Polytechnic National University

## THE CONCEPT OF CONVERTING TEXT INTO A BIT MAP USING ANY FONT

The modern world is impossible to imagine without the exchange of information. Newspapers, magazines, books, printed advertising products play an important role and are constantly being improved and require more and more resources and opportunities. At the moment, due to the sharp growth of digital technology, there is a problem between the capabilities of hardware and software. That is, when transferring data from the electronic world to paper, the difference between the computing capabilities of hardware and software is displayed. This applies to print media that cannot convey the full volume and depth of what is displayed on the screen. Raster fonts are used for printing faster, the main drawback of which is their non-scalability. When studying fonts in embedded systems, there were problems with the fact that there are quite a few ready-made options for converting text to a bitmap without certain restrictions. The font displayed on the screen and printed directly by the printer or other device may differ significantly because the number of pixels that can be printed is too large. Therefore, there is a need to optimize the conversion of the font into a bitmap, ie reducing the number of points without losing the base of the glyph. The paper presents the conversion of a font into a bitmap using a program based on a ruler, ie the division of the image into lines, which is used in printing. In order to create the required bitmap image in the program, you need to specify a specific glyph and its size, as well as the device for which it will be used. The scan converter takes a path and applies a set of rules to determine which pixels will be part of the glyph image when printed or displayed on the screen. The obtained results allow us to print the fonts we like, instead of the standard ones.

Keywords: fonts, bitmap, raster image, liniatura.

### Постановка проблеми

На сьогоднішній день існує велике різноманіття шрифтів, які використовуються у комп'ютерних системах. Однак, якщо мова заходить про використання шрифту у вбудованій системі (embedded system), то різноманітність різко зменшується до кількох варіантів. Тому існує велика проблема в динамічному перетворенні тексту в бітову карту з використанням того, чи іншого шрифту без жорстких обмежень. Бітова карта в цифрових зображеннях – це матриця, що зберігає значення елементів зображення (пікселів). Кожен піксель в складі растрового шрифту кодується за допомогою одного біта. Растрове зображення символу являє собою набір бітів – бітову карту. При відображенні тексту бітова карта символу копіюється на пристрій відображення. В наш час в мережі Інтернет доступно декілька варіантів форматів шрифтів. Проте, всі вони являються растровими. Коли для вбудованих систем це є недопустимим варіантом через те, що вони мають обмежену обчислювальну здатність. Подана проблема є актуальною. Перетворення растрового шрифту у бітову карту із заданою точністю (dpi) з використанням мінімальних ресурсів є актуальною проблемою сьогодні.

### Аналіз останніх джерел

Шрифт комп'ютерний – комплект букв, цифр і спеціальних символів, оформлених відповідно до єдиних вимог. Відрізняються малюнком (гарнітура), шрифтом (прямий, курсивний), розмірами (кегль). Апаратні механізми генерації літер певного накреслення при індикації їх на екрані монітора і при друку досить складні. Набір доступних користувачеві шрифтів залежить від того, які були інсталювані і які підтримуються пристроєм друку. Більшість шрифтів є масштабованими (scalable), що дозволяють без втрати

якості отримати будь-який розмір (кегль) даного накреслення.

Причому на жорсткому диску не зберігаються всі можливі кеглі даного шрифту. Вони генеруються (масштабуються) у міру необхідності (тобто обчислюються). Робота на персональних видавничих системах пов'язана з видачею символів на два пристрої: дисплей і принтер. У зв'язку з цим завжди слід уточнювати, про що мова: про принтерне або про екранне виведення. В обох випадках кожна літера представляється групою зафарбованих квадратиків (точок) на сітці шрифтового растра. Збільшений вигляд дає таку картину:

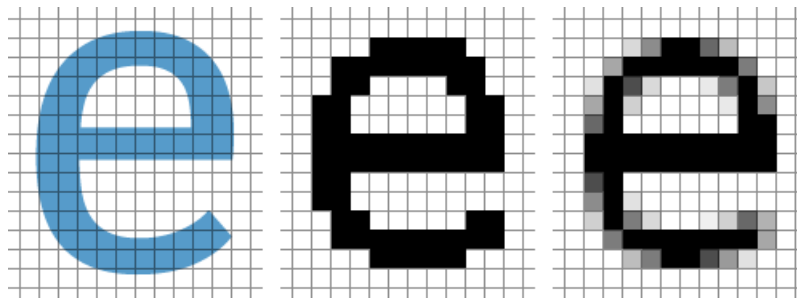


Рис. 1. Вид растрового символу «е» при збільшенні (набір точок)

Незважаючи на однаковий спосіб побудови, однойменні друковані та екранні шрифти відрізняються. Переважно розрізнення зводиться до того, що принтер приймає певний набір шрифтів, які не можна адекватно відобразити на екрані монітора. Доводиться робити пробну роздруку, що дає уявлення, чи задовольняє результат, або користуватися таблицями шрифтів.

Навіть якщо існує можливість адекватного відображення шрифту на дисплеї (за допомогою графічних редакторів або графічних режимів в текстових процесорах), не можна бути впевненим у повній відповідності зображення на екрані і на папері, через відмінні здібності принтера та дисплея. Розбігається також шкала щільності. Більшість принтерів сьогодні забезпечують роздільну здатність до 1200 крапок на дюйм. Але дозвіл звичайного монітора в середньому дорівнює приблизно 200 точок на дюйм (для екранів телефонів ця цифра зростає до 400 точок на дюйм). Це набагато гірше, ніж у «найслабшого» принтера. Тому текст на екрані часто сильно відрізняється від тексту на роздруковці.

Наступною причиною, що заслуговує уваги є відмінності в способах генерації символів для екранного шрифту та для принтера. Традиційним є метод бітової карти, коли генерується «вручну» один (або декілька) кеглів, а всі інші генеруються автоматично пропорційним збільшенням або зменшенням.

Збільшення не викликає проблем при цілочисловому коефіцієнті (2, 3 і т.д.): просто збільшується в відповідне число раз кількість точок на бітової матриці по вертикалі і по горизонталі. Набагато складніше виконувати збільшення при дрібних коефіцієнтах. З труднощами пов'язане також зменшення кегля. Але навіть збільшення в ціле число раз «не відбувається гладко». Стає яскраво вираженим так званий «сходовий ефект»: дрібні сходишки з крапок, з яких утворюється символ.

З деяких пір всі розроблені шрифти генеруються не на основі бітових карт, а шляхом побудови ескізів літер набором описуваних формулами кривих. У момент друку за формулами будується бітова карта для будь-якої роздільної здатності і розміру.

Математичний опис форм літер забезпечує друкарську якість і уникнення «сходового ефекту» для зображення для будь-яких кеглів. Особливо потужним засобом є спеціальна мова опису сторінок (мова програмування) PostScript, що дозволяє робити над символами і малюнками різноманітні маніпуляції: обертання, розтягнення, стиснення і т.д. Подання літер, що генеруються відповідно до математичних формул, однакове на будь-якому пристрої. З точки зору сучасних комп'ютерних систем втрачає силу старе правило, згідно з яким набір доступних шрифтів визначається використовуваним принтером.

**Метою роботи** є розробка програмного забезпечення для перетворення растрових шрифтів у бітову карту задля друку тексту відповідним шрифтом із заданою точністю.

#### Виклад основного матеріалу

Для досягнення поставленої мети було вирішено розробити програмне забезпечення на мові C, що буде перетворювати шрифт у бітову карту. Для цього потрібно зрозуміти схему формування шрифту на екрані та його підготовку для відображення.

Існує два типи шрифтів для пристроїв, які можна вказати на принтерах:

- Растрові шрифти
- Масштабовані шрифти

Растрові шрифти – це шрифти, які мають визначення гліфу (в технологіях введення тексту, гліф – це сімейство методів введення тексту, засноване на розбитті букв на базові форми) фіксованого розміру. Масштабовані шрифти – це шрифти, які мають у принтері математичне визначення, щоб їх можна було намалювати будь-якого розміру. Тобто його розмір змінюється.

Класичним прикладом растрового шрифту є Courier Characters per Inch (IRC). Як впливає з назви, цей шрифт є переходом від друкарських машинок до принтерів ударного типу. Його називають растровим шрифтом, оскільки найпоширенішим визначенням шрифту було відображення растрового зображення ПЗУ з матричного принтера.

На більшості принтерів з PostScript (мова опису сторінок, в основному використовується в настільних видавничих системах) можна знайти приклади масштабованих шрифтів, що постійно розміщуються на принтері, де зазвичай існує стандартний набір шрифтів PostScript, таких як Helvetica та Times.

Растрові зображення зберігаються у вигляді серії крихітних точок, які називаються пікселями. Кожен піксель насправді є дуже маленьким квадратиком, якому присвоюється колір, а потім розташовується у візерунку для формування зображення. При збільшенні растрового зображення ви можете побачити окремі пікселі, що складають це зображення.

Растрова функція – двомірна квадратна матриця, в осередках якої стоять вагові значення субелементів, що утворюють друковані або пробільні елементи. Функції растра дозволяють задавати операції обробки, які можуть застосовуватися до одного або декількох растрів. Ці функції застосовуються до растрових даних «на льоту» під час доступу до зображень і їх перегляду.

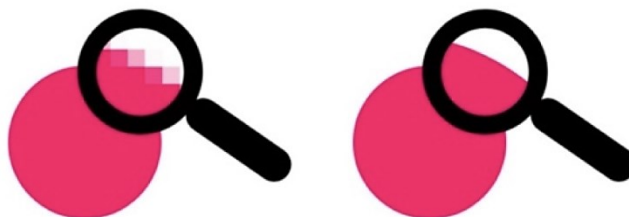


Рис. 2. Растрове та векторне зображення при масштабуванні

Під час друку зображення на принтері враховується ще одна важлива функція, а саме частота растра. При раструванні, зображення представляється у вигляді матриці півтонових осередків. Розмір напівтонового осередку матриці, що визначає лініатуру (lpi), може змінюватися, а ось відстань між друкованими точками фіксована і залежить від дозволу принтера (dpi). Частота растра, або лініатура растра, параметр, що характеризує растрову структуру кількістю її елементів (ліній з растрових точок) на одиницю довжини. В даний час лініатуру вимірюють в лініях на дюйм.

Головна проблема, що стоїть перед нами є те, що растрові шрифти мають зовелику роздільну здатність і тому не можуть відтворюватися у вбудованих системах. Тобто кількість пікселів, які мають бути надрукованими є зовеликою. Саме тому, основною ідеєю програми є зменшення кількості точок без втрати основи гліфу.

При розробці програмного забезпечення, за основу, було взято лініатуру, тобто поділ зображення на лінії, який використовується у поліграфії. Коли програма запитує певний гліф певного розміру для певного пристрою, механізм шрифтів створить необхідне растрове зображення. Першим кроком у цьому процесі є масштабування основного контуру до бажаного розміру. Масштабування контуру – це робота тієї частини механізму шрифтів, відомого як скалер. Коли основний контур масштабується, точки, що складають контур гліфа, змінюються з незалежних від пристрою одиниць вимірювання. Після того, як основний контур буде масштабований, він готовий до растризації за допомогою перетворювача сканування. Перетворювач сканування приймає контур і застосовує набір правил, щоб визначити, які пікселі будуть частиною зображення гліфа при друку або відображенні на екрані. Після обробки програмою виходить результат наведений на рисунку нижче:

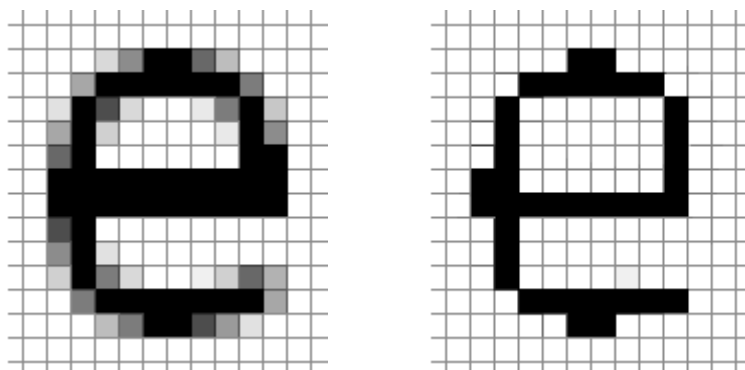


Рис. 3. Символ до обробки програмою (ліворуч) та після її використання (праворуч)

За рахунок зменшення частоти растра є можливість друкувати «об'ємні» шрифти у вбудованих системах.

### Висновки

Для друку тексту відповідним шрифтом із заданою точністю розроблено програмне забезпечення. Воно дозволяє друкувати вибрані шрифти у вбудованих системах не перетворюючи їх на «примітивні» вбудовані у пристрої стандартні шрифти. Ця проблема є вузько направленою та вимагає істотних

досліджень в області поліграфії та друкарства. За рахунок опрацювання даних областей науки і було розроблено програмне забезпечення.

### Література

1. Філіпович А.Ю. Розробка шрифтових елементів : курс практичних занять / Філіпович А.Ю. – 2008. – 129 с.
2. Шрифти. Розробка і використання / [Баришніков Г.М., Бізя А.Ю., Єфімов В.В., Моїсеєв А.А., Почтар Е.І., Ярмола Ю.А.]. – Москва : Друкарня АО «Молода гвардія».
3. V.A. Artemov (1933) Technographic analysis of summarized letters of new alphabet. Writing and revolution. 1, 58–76.
4. Lilley C. The "font" Top-Level Media Type. Internet Engineering Task Force, 2017. 18 p.
5. Murray, Stephen. "Graphic Devices." Computer Sciences, edited by Roger R. Flynn, vol. 2: Software and Hardware, Macmillan Reference USA, 2002, pp. 81–83.
6. Анісімов В.А. Основні алгоритми обчислювальної геометрії : навч. посібник / Анісімов В.А., Терещенко В.М., Кравченко І.В. – К. : Київський університет, 2002. – 82 с.
7. Ньюмен У. Основы интерактивной машинной графики / Ньюмен У., Спрул Р. – М. : Мир, 1976. – 574 с.
8. Шикин Е.В. Кривые и поверхности на экране компьютера / Шикин Е.В., Плисс А.И. – М. : Диалог МИФИ, 1996. – 240 с.
9. Гайдуков С.А. OpenGL. Профессиональное программирование трехмерной графики на C++ / Гайдуков С.А. – СПб : Питер, 2004. – 716 с.

### References

1. Filipovich A.Y. Development of font elements: a course of practical training. 2008. p. 129
2. Baryshnikov G.M., Bizya A.Yu., Yefimov V.B., Moiseev A.A., Pochtars E.I., Yarmola Y.A. Fonts. Development and use, Printing House AO "Young Guard", Moscow
3. V.A. Artemov (1933) Technographic analysis of summarized letters of new alphabet. Writing and revolution. 1, 58–76
4. Lilley C. The "font" Top-Level Media Type (English) - Internet Engineering Task Force, 2017. 18 p.
5. Murray, Stephen. "Graphic Devices." Computer Sciences, edited by Roger R. Flynn, vol. 2: Software and Hardware, Macmillan Reference USA, 2002, pp. 81–83.
6. Anisimov V.A., Tereshchenko V.M., Kravchenko I.V. Basic algorithms of computational geometry: Textbook. Manual. Kyiv: Kyiv University, 2002. 82 p.
7. Newman W., Sprul R. Fundamentals of interactive machine graphics. M.: Mir, 1976. 574 p.
8. Shikin E.V., Pliss A.I. Curves and surfaces on a computer screen. M.: Dialogue MEFH, 1996. 240 p.
9. Gaidukov S.A. OpenGL. Professional programming of 3D graphics in C ++. SPb: Peter, 2004. 716 p.

Надійшла/Paper received : 08.03.2021 р. Надрукована/Printed : 02.06.2021 р.