طراحی خودکار حروف خط نستعلیق*

محمّد قدسی **کیارش بازرگان**

دانشکدهٔ مهندسی کامپیوتر
دانشگاه صنعتی شریف

چکیده

در این مقاله یک روش سریع و خودکار برای طراحیِ «علمیِ» حروف مختلف خط نستعلیق ارائه می‌شود. منظور از طراحیِ علمیِ حروف، استفاده از منحنی بِزیه و پارامترهای مختلف در طراحی است به طوری که بتوان اندازه‌های مختلف یک حرف و سبک‌های متفاوت نگارش آن را با کیفیت عالی و تنها با تغییر مقادیر چند پارامتر به سرعت تولید کرد. مِتافونت، زبانی برای طراحی علمی فونت‌ها است که در این مقاله از آن استفاده می‌شود. بااستفاده از امکان «قلم» در این زبان، می‌توان حرکت قلم‌درشتِ یک استاد خوشنویس را شبیه‌سازی کرد. استفاده مستقیم از متافونت برای طراحی کیفی فونت‌ها کاری طولانی و همراه با مشکلات متعددی است. در این مقاله نرم‌افزار معرفی می‌شود که به کمک آن تصویر پویش شده یک حرف به صورت خودکار به برنامهٔ متافونت جهت تولید آن فونت تبدیل می شود.

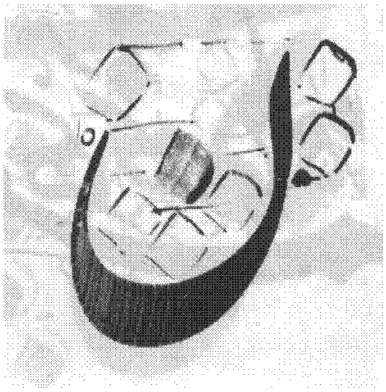
۱ معرفی

خط نستعلیق فارسی یکی از زیباترین هنرها در کشور ماست. هرچند ورود کامپیوتر به حیطهٔ این هنر باید با احتیاط، دقت و ظرافت صورت گیرد، ولی استفاده از این ابزار، که تولید فونت خط نستعلیق با کیفیت عالی و نیز حروفچینی سریع با این خط را ممکن می‌سازد، اجتناب ناپذیر است.

یکی از مراحل مهم تولید هر سیستم حروفچینی فارسی، طراحی مجموعه کاملی از فونتهای مختلف حروف فارسی با کیفیت عالی است. این کار معمولاً بسیار وقت‌گیر و انجام صحیح آن بسیار دشوار است. اهمیت و سختی این کار برای خط نستعلیق بسیار واضح است.

با استفاده از ابزارهای کامپیوتری و روش‌های های مبتنی بر اصول طراحی هندسی می‌توان کیفیت و سرعت

*ارائه شده در اولین کنفرانس سالانهٔ انجمن کامپیوتر ایران، دی‌ماه ۱۳۷٦، دانشگاه صنعتی شریف، تهران.
[1] METAFONT



شکل ۱: تصویر پویش شده نویسه «ن».

طراحی را بالا برد. متافونت[۱] [۱] زبانی است عمدتاً برای طراحی فونت‌ها که در سال ۱۹۸۲ توسط دانلد کنوث[۲]، استاد دانشگاه استانفود آمریکا، برای طراحی قلم‌های استفاده شده در سیستم حروفچینی TEX ابداع شد [۳]. با این زبان می‌توان برنامه‌هایی حاوی دستورها و پارامترهای مختلف نوشت که با اجرای آنها، حروف طراحی شده تولید خواهد شد.

متافونت در طراحی خود از منحنی‌های بزیه[۳] استفاده می‌کند، و این از محاسن آن است. چون، فونتهای طراحی شده با آن قابلیت «مقیاس شدن»[۴] دارند و می توان به سادگی و با سرعت آن‌ها را در اندازه‌های مختلف و با کیفیت عالی ایجاد کرد. همچنین می‌توان طراحی را به صورتی انجام داد تا با تغییر مقادیر چند پارامتر، مجموعه‌های مختلف و هم‌خانواده از فونت‌ها (مثلاً، سیاه، ایتالیک، ایرانیک، و غیره) به سادگی تولید شوند. به عنوان مثال، اثرات تغییر پارامتر «کشش» در حرف «ب» در شکل ۲ نشان داده شده است.

همچنین در این زبان با استفاده از امکانی به نام «قلم» و دستورهای مناسب، می‌توان حرکت قلم‌درشتِ یک استاد خوش نویس را شبیه سازی نمود. برای استفاده از این امکان، ابتدا باید چند «مکان قلم[۵]» (شامل ضخامت، زاویه قرارگرفتن قلم در این نقطه) را مشخص کرد. سپس با تعیین ترتیب عبور، زاویهٔ حرکت و میزان کشش قلم در این مکان‌ها شکل حرف را توصیف نمود[۶].

مزیت دیگر زبان متافونت نسبت به ابزارهای دیگرِ طراحی حروف، استفاده از پارامترهای سطح بالا و در عین حال پایدار است. تغییر مقادیر این پارامترها، باعث تغییر ناگهانی شکل حروف نمی‌شود (در حالی که، سیستم‌هایی که منحنی مرزی فونت را به وسیله قطعه—تابع‌های درجه ۳ تعریف می‌کنند، نسبت به نقاط کنترلی این تابع‌ها، حساس هستند.) علاوه بر این‌ها، قطعه‌های مشترک بین چند حرف از یک یا چند فونت را نیز می‌توان فقط یکبار به صورت مجزا طراحی نمود. از اتصال این قطعات به هم می‌توان حروف مختلف را به سادگی تولید کرد. در طراحی این قطعات می‌توان سبک‌های نوشتاری اساتید خوشنویس را با تعریف پارامترهایی توصیف نمود. با مقداردهی مناسب به این پارامترها، حروف به صورت همگن و با سبک‌های نوشتاری مختلف قابل تولید هستند.

البته نوشتن یک برنامه متافونت با ویژگی‌های فوق به روش دستی و سعی و خطا، به خصوص برای تولید فونت‌های خط نستعلیق، کار بسیار دشواری

---

[۱] Metafont (footnote not visible)

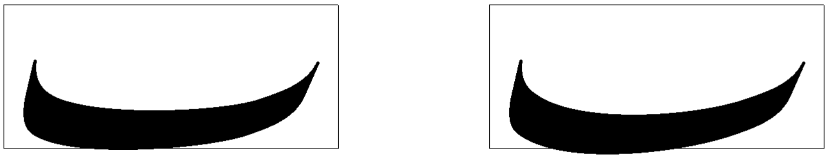
[۲] Donald E. Knuth

[۳] Bézier

[۴] scalability

[۵] penpos

[۶] در ادامه این مقاله، منظور ما از «قلم» همین امکان متافونت است و از «ویژگی‌های قلم» مشخصات ذکر شده در این پاراگراف است.



شکل ۲: اثرات تغییر پارامتر «کشش» در حرف «ب».

است. در این مقاله، یک محیط نرم‌افزاری محاوره‌ای[۷] برای طراحی خودکار فونت‌ها ارائه می‌شود که در آن با استفاده از تصویر پویش شده[۸] یک حرف و با دریافت اطلاعات مختلف از کاربر بصورت گرافیکی، برنامه متافونت مورد نیاز برای تولید آن حرف به صورت خودکار تولید می‌شود. ما از این سیستم در طراحی حروف خط نستعلیق استفاده کرده‌ایم.

در این مقاله ابتدا به معرفی قدم‌های لازم برای طراحی خودکار فونت‌ها می‌پردازیم و سپس روش‌های مورد استفاده در سیستم موجود را بررسی کرده و در انتها روش‌های گسترش و بهبود آن را معرفی می‌نماییم.

۲ اهداف

هدف ما تولید نرم افزاری است که با حداقل محاوره با کاربر بتواند تصویر پویش شده یک حرف را دریافت کرده، اعمال زیر را انجام دهد:

۱) نویز تصویر پویش شده حرف را طوری حذف نماید که ضمن حفظ جزئیات مهم شکل آن، نوسانات ریز منحنی مرزی از بین برود.

۲) منحنی مرزی حرف را با مجموعه‌ای مناسب از قطعات منحنی بزیه تقریب بزند. منظور از «مجموعه مناسب» مجموعه‌ای از این منحنی‌ها است که:

الف) تقریب بهینه‌ای از مرز حرف باشد و مشخصات ظاهری حرف (مثل انحنا یا زاویه خط مماس بر منحنی در هر نقطه) را حفظ کند.

ب) توسط قطعه—منحنی‌ها بتوان به راحتی ویژگیهای قلمِ متافونت را بدست آورد.

۳) تبدیل منحنی‌های بزیه به ویژگیهای قلم، به طوریکه شکل حرف نسبت به تغییر مقادیر این پارامترها حساس نباشد. مکان‌های قلم باید در بحرانی‌ترین نقاط حرف قرار گرفته باشند و تعداد آنها حداقل باشد.

۴) دسته‌بندی قطعه—منحنی‌های مرزی حروف از یک یا چند فونت به مجموعه‌هایی از منحنی‌های «مشابه» و توصیف هر مجموعه با تعدادی پارامتر و ویژگی‌های قلم به طوریکه بتوان با تغییر این پارمترها، هریک از منحنی‌های این مجموعه را بدست آورد. با این کار، یک مجموعه از منحنی‌های مشابه تنها با یک منحنی و تعدادی پارامتر قابل توصیف خواهند شد. لذا، تعداد کل منحنی‌ها به شدت کاهش یافته، می‌توان آنها را با دقت بیشتری طراحی کرد. با استفاده از این روش می‌توان قطعات حروف را با مقداردهی مناسب به پارامترها تولید کرد و با متصل نمودن تعدادی قطعه—حرف، حروف کامل را حتی در فونت‌های مختلف (سیاه، نازک و غیره) ایجاد کرد.

---

[۷] interactive
[۸] scan

شکل ۳: الف: منحنی‌های مرزی حرف «ن» پس از اجرای الگوریتم کانتور   ب: پس از فیلتر شدن.

۳ روش‌های استفاده شده

سیستم موجود شامل مجموعه‌ای از ماکروها به زبان متافونت و یک محیط محاوره‌ای گرافیکی است. پس از تهیه تصویر پویش شدهٔ حروف مختلف، کاربر می‌تواند با استفاده از امکانات مختلف سیستم، به سادگی حروف مختلف را از هم جدا ساخته و نویز موجود را حذف نماید و ویژگی‌های مختلف حروف را اندازه‌گیری نماید. کاربر سپس با روش‌هایی که در زیر می‌آید می‌تواند برنامهٔ متافونت هر یک از حروف را به صورت خودکار و با کیفیت عالی تولید کند. با اجرای این برنامه‌ها، فونت‌های مختلفی از آن حرف ایجاد خواهد شد. در ادامهٔ این بخش، روش‌های موجود برای انجام امور ذکر شده در بخش ۲ خواهد آمد.

در سیستم موجود، حروف با دقت بالا (DPI ۳۰۰-۶۰۰) و به صورت Gray Scale پویش می‌شوند و یک حرف به تصویری به ابعاد ۷۰۰ × ۷۰۰ تبدیل می‌گردد. برای حذف نویز این تصاویر، سه روش را بررسی کرده‌ایم و روش سوم را به دلائلی که ذکر می‌شود، انتخاب نموده‌ایم.

در روش اول، ابتدا تبدیل فوریه گسسته[۹] [۵] تصویر را بدست می‌آوریم. سپس آنرا از یک فیلتر پایین‌گذر ایده‌آل دو بعدی[۱۰] [۴] عبور داده و IDFT آنرا محاسبه می‌نماییم. در انتها منحنی‌های مرزی تصویر را بدست می‌آوریم. اِشکال عمده این روش (غیر از کندی ذاتی آن) این است که نمی‌توان فرکانس قطع فیلتر را برای قسمتهای مختلف تصویر تغییر داد و اغلب باعث حذف ناخواسته جزییات منحنی می‌شود.

در روشهای بعدی، ابتدا با استفاده از الگوریتم «کانتور» با همسایگی ۴ نقطه‌ای (معمولاً با آستانه تغییر روشنایی بین ۷۰٪ و ۵۰٪) مرز حروف را بدست می‌آوریم و سپس نویز این منحنی‌ها را حذف می‌کنیم.

در روش دوم، منحنی بستهٔ مرزی حرف را به مجموعه‌ای از L نقطه $z = x + jy$ تبدیل می‌کنیم، بطوریکه برای $t = 0, 1, \ldots, L - 1$، $x = f(t)$ و $y = g(t)$ باشد. سپس z را فیلتر می‌کنیم. برای فیلتر کردن z، توابع غیر خطی $A(t) = arg(z)$ و $M(t) = mag(z)$ را جداگانه فیلتر کرده و توابع f و g را از روی آنها بازسازی می‌کنیم. منحنی حاصل، یک منحنی پیوسته و فاقد نوسانات ناشی از نویز خواهد بود و در عین حال، جزئیات منحنی را به حد مطلوبی حفظ می‌نماید. این روش در مقایسه با روشهای دیگر فیلتر کردن بسیار سریع می‌باشد.

روش آخر که در این مقاله مورد استفاده قرار می‌گیرد، از مقاله [۷] اقتباس شده است که در آن، تصویر به روش Scale Space فیلتر شده و پس از گذراندن از یک فیلتر گاوسی[۱۱]، منحنی مرزی شکل بصورت کنترل شده‌ای صاف می‌شود. مزیت این روش به روش قبل

[۹] Discrete Fourier Transform (DFT)

[۱۰] 2-d ILPF

[۱۱] Gaussian

این است که خطای کوانتیزه شدن منحنی پویش شده را بهتر از بین می‌برد و در نتیجه برای عملیات بعدی روی منحنی، از جمله مشتق گیری، نیازی به درونیابی نقاط نیست.

در سیستم موجود برای تقریب منحنی مرزی حرف با منحنی های بزیه و تعیین ویژگیهای قلم، به دو صورت می‌توان عمل کرد:

۱) کاربر بصورت گرافیکی ویژگیهای قلم را در چند نقطه از مسیر حرکت تعریف می‌کند و همزمان، در پنجره‌ای دیگر، برنامهٔ متافونت آن حرف بطور خودکار تولید می‌شود. در صورت لزوم کاربر می‌تواند این برنامه را ویرایش کند. در این روش، تعیین ویژگیهای قلم عمدتاً مبتنی بر تجربیات کاربر است. شکل ۴—الف بدین روش تولید شده است.

۲) کاربر یک قطعه از منحنی را انتخاب می‌کند و برنامه توسط یک روش مکاشفه‌ای [۱۲]، منحنی انتخاب شده را با بهترین منحنی بزیه ممکن (با حداقل مجموع مربع اختلاف نقاط منحنی بزیه و منحنی اصلی) تقریب می‌زند. یک منحنی بزیه را می‌توان بصورت دقیق با تعیین نقاط شروع و پایان و نیز بردار سرعت منحنی در این نقاط تعریف نمود [٦]. در ابتدا، بردار سرعت در نقاط شروع و پایان تقریب زده می‌شود و سپس با تکرار این روش، منحنی مورد نظر با دقت کافی بدست خواهد آمد. مجموعه منحنی‌های تقریب زده شده، توسط یک ماکرو که به زبان متافونت نوشته شده است، به حرکت قلم تبدیل می‌شود. کاربر می‌تواند مکان و ویژگی قلم‌ها را به دلخواه، تغییر دهد. حرف «ع» نشان داده شده در شکل ۴—ب با این روش طراحی شده است.

در سیستم طراحی شده، همچنین می‌توان حرکتهای مشابه قلمِ متافونت را در بخش‌هایی از حروف مختلف بدست آوَرد. با این امکانات، کاربر منحنی‌های مشترک را از حروف جدا می‌کند و آنها را بصورت مجزا تولید می‌نماید. از بهم‌چسباندن این قطعات، حروف نهایی تولید می‌شوند.

۴ نتیجه‌گیری

در این مقاله یک سیستم نرم افزاری برای تولید خودکار فونت‌ها به زبان متافونت ارائه گردید. به دلیل ویژگیهای متافونت، سیستم موجود می تواند برای طراحی سریع و با کیفیت عالی حروف خط نستعلیق استفاده شود. در سیستم طراحی شده، تصویر پویش شده یک یا چند حرف با استفاده از ابزارهای گرافیکی فراهم شده به صورت خودکار به برنامه متافونت تبدیل می شود. این کار، با حذف نویز از تصویر، تقریب منحنی مرزی با قطعات منحنی بزیه و سپس تعیین مشخصات قلم متافونت انجام می‌شود. هر چند که این کار در حال حاضر با کیفیت قابل قبولی انجام می‌شود و سیستم موجود عملاً می تواند زمان طراحی یک مجموعه فونت را از ماهها به چند هفته کاهش دهد، ولی کار برای تکمیل مراحل خودکار سازی ادامه دارد. در این راستا، ایده‌های مطرح شده در [۷] می‌تواند موثر باشد. ضمناً ایده‌های مطرح شده در این مقاله در تشخیص نوری کلمات و حروف (OCR) نیز می تواند مورد استفاده قرار گیرد. پروژه‌هائی هم در این زمینه در دست اجرا است.

[۱۲] heuristic

شکل ۴: الف: حرف «ن» تولید شده   ب: تعیین مشخصات حرکت قلم در حرف «ع».

مراجع

[1] D. E. Knuth, *The* METAFONT*book*, Addison-Wesley, 1985.

[2] D. E. Knuth, METAFONT*: The program*, Addison-Wesley, 1985.

[3] D. E. Knuth, *Computers and Typesetting*, Addison-Wesley, 1986.

[4] A.V. Oppenheim, A.S. Willsky, *Signals and Systems*, Prentice-Hall Innternational, 1983.

[5] R.C. Gonzalez, P. Wintz, *Digital Image Processing*, Addison-Wesley, 1987 2nd ed.

[6] J.D. Foley, A. van Dam, S.K. Feiner, *Computer Graphics: Principles and Practice*, Addison-Wesley, 1990, 2nd ed, 478-491.

[7] K.Xin, B.Lim, G.S. Hong "A Scale-Space Filtering Approach For Visual Feature Extraction", Pattern Recognition Journal, Vol 28 #8, Aug 1995, 1145-1158.