Modifying PDF Sewing Patterns for Use With Projectors

Charlotte Curtis
ccurtis@mtroyal.ca
Mount Royal University
Calgary, Alberta, Canada

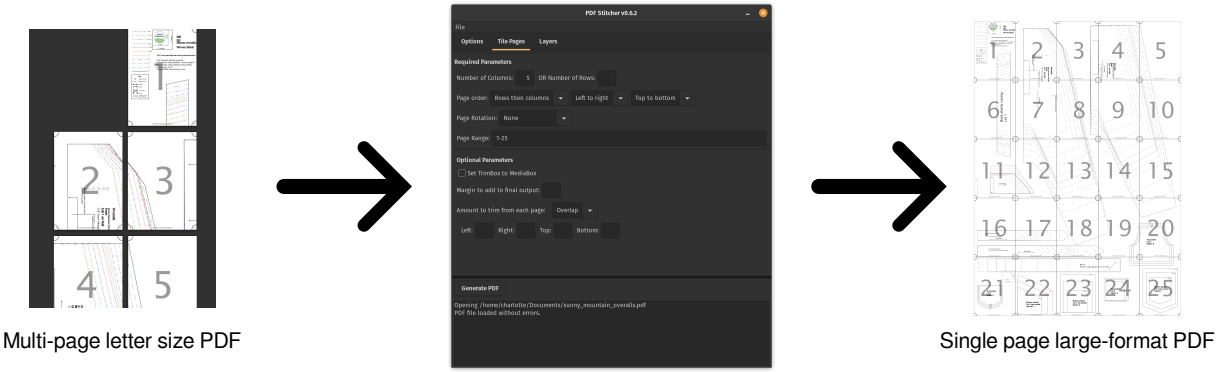

Figure 1: Performing page imposition of a typical sewing pattern with PDFStitcher. Sample pattern used with permission from Cathleen Hutchins of Sunny Mountain Patterns.

ABSTRACT

Print-at-home PDF sewing patterns have gained popularity over the last decade and now represent a significant proportion of the home sewing pattern market. Recently, an all-digital workflow has emerged through the use of ceiling-mounted projectors, allowing for patterns to be projected directly onto fabric. However, PDF patterns produced for printing are not suitable for projecting.

This paper presents PDFStitcher, an open-source cross-platform graphical tool that enables end users to modify PDF sewing patterns for use with a projector. The key functionality of PDFStitcher is described, followed by a brief discussion on the future of sewing pattern file formats and information processing.

CCS CONCEPTS

• **Information systems** → Open source software; **Document structure**; Information extraction; • **Applied computing** → Media arts; *Format and notation.*

KEYWORDS

PDF, document transformation, sewing patterns

ACM Reference Format:
Charlotte Curtis. 2022. Modifying PDF Sewing Patterns for Use With Projectors. In *DocEng '22: Proceedings of the 22nd ACM Symposium on Document Engineering.* ACM, New York, NY, USA, 4 pages.
https://doi.org/10.1145/3558100.3563853

DocEng '22, September 20–23, 2022, Virtual Event, CA, USA

ACM 978-1-4503-9544-1/22/09…$15.00
https://doi.org/10.1145/3558100.3563853

1 INTRODUCTION

Like many other fields, PDF has become the *de facto* standard for digital sewing pattern distribution. People who produce, purchase, and use sewing patterns represent a diverse group of needs and usage habits, with a number of common core elements. PDFStitcher, an open-source cross-platform graphical tool has been designed to accommodate these differing needs, with a particular focus on projector sewing.

1.1 Background

Home sewing patterns are documents typically consisting of two parts: a set of 2D shapes to cut from fabric, and the instructions on how to assemble the flat shapes into a 3D garment or other item. As the former requires paper sizes of A0 or larger, such patterns were traditionally published on large-format tissue paper, limiting production capability to only a few large companies [5].

A resurgence in sewing interest, combined with the advent of the internet, has allowed for a new cottage industry of independent sewing pattern designers [6]. So-called "PDF patterns" are sold as digital downloads that end users can print at home on standard letter or A4 paper, then tape together to create a large format page before cutting out the individual pattern pieces. While the scale of this market is difficult to estimate, a recent survey suggested that a majority of home sewers or sewists[1], with participants from up to 20% of US households, prefer independent PDF patterns [7].

It is equally difficult to estimate the number of independent pattern designers. A popular review website lists 940 distinct companies, but it is likely that this is not comprehensive. Each of these designers produces patterns in their own style, but a common approach is to draw the pattern in vector graphics software such as

[1] In keeping with trends observed in the online sewing community, "sewist" has been adopted in this paper as the preferred term describing people who sew.

Adobe Illustrator and generate a PDF containing multiple clothing sizes, with each size denoted by a distinct line style and contained in a "layer" or PDF optional content group (OCG) [8]. The leftmost image in figure 1 shows the first few pages of a typical print-at-home sewing pattern.

While PDF patterns marked a significant change in the home sewing pattern industry, a new trend of paperless sewing is emerging through the use of digital projectors, allowing for pattern pieces to be projected onto the fabric and cut directly. A patent for such a process was filed in December 2017 [2], but to date, no commercial product has been released. Instead, home sewists have begun developing their own processes, primarily discussed and shared in the FaceBook group "Projectors for Sewing". Started by Missy Pore in late 2019, this group now has over 53k members, suggesting a strong interest in projector sewing.

Section 2 presents an overview of PDFStitcher's functionality and interface, while Section 3 provides a comparison to similar tools, evidence of community impact, and a brief speculation on the future of sewing pattern document analysis and file formats.

1.2 Motivation

Many independent PDF pattern designers are incorporating projector files into their digital products, typically consisting of large-format pages with bold lines that are readily visible on the inexpensive 720p projectors used by most projector sewists. However, the majority of offerings are still produced as print-at-home PDFs, with pattern pieces split across multiple letter/A4 size pages.

In addition to page format differences, projector users often want to add margins to allow for greater positioning flexibility, change line styles to accommodate different fabric patterns, and remove layers to import a single size into a vector drawing program such as Inkscape for further modification.

2 SOFTWARE OVERVIEW

PDFStitcher is written in Python 3.8 using PikePDF to read and write PDF contents and wxPython for cross-platform GUI components. PDFStitcher is published on PyPi and Flathub, while Windows and macOS binaries are produced using PyInstaller and available for download from GitHub.

As of v0.7.1, PDFStitcher provides the following functionality:

- Page imposition (N-up) with optional page trimming, overlap, or rotation.
- Adding margins to single pages or after imposition.
- Layer selection with options to remove or hide content, including "non-optional" content or locked layers.
- Modification of line properties for one or multiple layers.

Figure 2 shows the interface presented to the user upon launch. All screenshots taken from Pop!_OS 22.04 running the default GNOME desktop environment.

2.1 Page Imposition

The core functionality of PDFStitcher is to "stitch together" PDF pages into a single large format page suitable for projecting, without modifying the document scale. Figure 1 shows the overall process with a representative sample sewing pattern as well as the "Tile Pages" tab of the PDFStitcher interface.

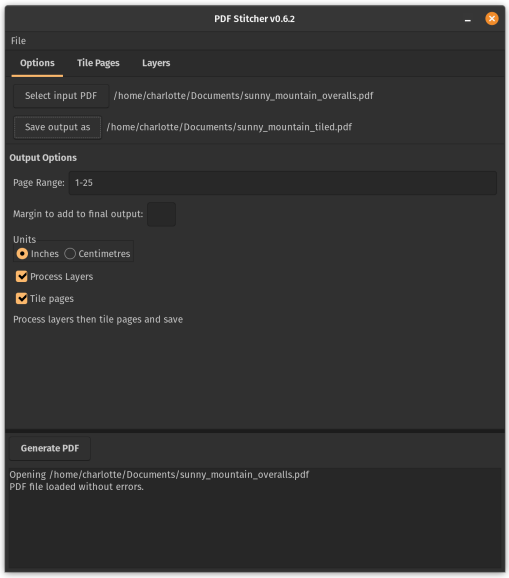

Figure 2: The main Options tab of PDFStitcher.

While the sample pattern shown is a straightforward alignment of pages that do not require trimming, many patterns are intended to have a small margin trimmed or overlapped, as home printers do not print to the edge of the paper. PDFStitcher allows for this by providing text boxes for the user to enter a trim value in inches or centimetres for each side of the page.

Additional options include adding a margin around the final single-page result for greater scrolling flexibility, specifying page ranges with optional "0" to insert a blank page, defining the grid by rows or columns, top to bottom or left to right, and page rotation.

Page imposition is accomplished by converting each page into a form external object (XObject), then placing each one on a new page with dimensions calculated from the various options. If "overlap" is selected the XObjects are placed unmodified with the position adjusted by the user-specified overlap setting, while if "trim" is selected the XObject TrimBox is modified first.

2.2 Layer Manipulation

Projecting an image results in lower resolution and lower contrast than printing the equivalent image, particularly with the low-cost LED projectors favoured by projector sewists. This leads to a need for thicker cutting lines and the ability to change colour based on fabric colour. Furthermore, many projector sewists want to remove unused size layers to make adjustments to the pattern pieces in a vector graphics program such as Inkscape. Inkscape can import PDF graphics, but has no awareness of PDF layers, so isolating the layer of interest in PDFStitcher first simplifies the vector graphic editing process.

Figure 3 shows the "Layers" tab presented to the user. Each of the OCGs in the PDF are listed with checkboxes to include/exclude in the final output, as well as line property override options. Setting the line colour, thickness, or style modifies all of the vector graphics found in the selected layer. Figure 4 shows the sample pattern with

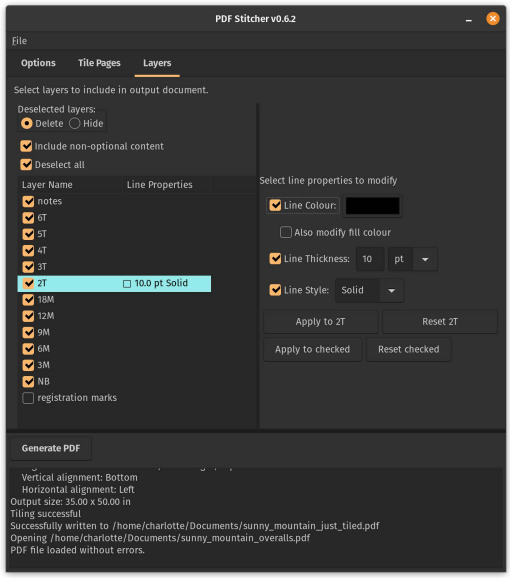

Figure 3: The Layers tab of PDFStitcher.

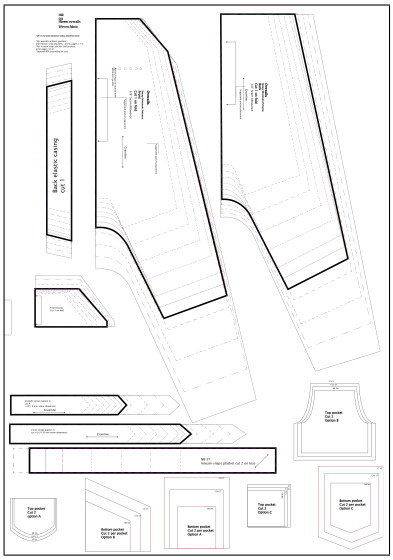

Figure 4: Sample pattern after applying layer modifications.

the 2T size modified to be a 10 pt solid black line, and with the "registration marks" layer removed for easier viewing.

In addition to modifying, disabling, or removing OCGs, PDFStitcher also provides the ability to remove "non-optional content" or locked layers, allowing for removal of content that cannot be hidden in Adobe Reader or other PDF viewers.

Layer manipulation is performed by parsing the PDF content stream found in pages and form XObjects. When an OCG selected for removal is encountered, the PDF "showing" operators (path-painting, text-showing, XObject placement, etc) and operands are excluded from the stream; all other operators related to the graphics state are retained [8]. XTable 1 summarizes the operators removed from the stream. Empty `q`/`Q` blocks are also removed to reduce content stream clutter.

Table 1: PDF operators excluded from the content stream when layers selected for removal

Category	PDF Operators
Path-painting	`b, B, b*, B*, c, f, F, f*, h, l, re, s, S, sh, v, y`
Text-showing	`Tj, TJ, ', "`
Other	`BI/ID/EI, Do`

Objects labelled with OCGs or optional content membership dictionaries (OCMDs) are treated as if they were invoked within the corresponding OCG in the page content stream.

To modify line style properties the graphics state is tracked internally while parsing the content stream. When a path modification operator is encountered, the operands are replaced by the desired properties. When a path-painting operator is encountered, the internal state is checked. If it differs from the desired state, the appropriate path property operator is inserted before the painting command. Finally, when the end of the OCG is reached, the previous state is restored even if no `Q` command is issued. Care is also taken to ensure the state is preserved when placing form XObjects within a content stream; this may also occur recursively.

3 DISCUSSION

3.1 Comparison to Similar Tools

Prior to the release of PDFStitcher, sewists were using a number of manual processes to perform page trimming and imposition, such as importing and aligning one page at a time in Inkscape. While a patent entitled "Method of Formatting Sewing Patterns for Paperless Use" was filed in 2019 [1], to date, no such method has been made available to the public, and the described process contains little detail on proposed implementation.

Several online PDF manipulation services provide N-up capability, such as PDFResizer.com, sejda.com, and deftPDF.com. However, these services are intended for producing booklets, providing limited page layout choices without preserving document scale. Additionally, transmitting purchased PDFs to a third party service is likely to violate the usage agreement between the sewist and the pattern designer.

Local tools, such as CutePDF Professional or VeryPDF PDF Stitcher[2], may overcome these limitations, but do not provide the additional functionality of line property and layer manipulation.

The free Adobe Reader DC provides the ability to modify displayed vector colours, but this is performed *en masse* rather than affecting a single layer. Vector graphics software such as the open-source Inkscape or paid Adobe Illustrator allow for selective line property modification, but this is a tedious manual process and the final result must be re-exported back to PDF. To the best of our knowledge, there is no alternative to PDFStitcher for modifying the line properties of a PDF on a layer-by-layer basis.

[2]VeryPDF PDF Stitcher was named before the PDFStitcher described in this paper.

3.2 Community Impact

PDFStitcher began as a simple Python script for page imposition. After sharing with the Projectors for Sewing FaceBook community, a graphical interface was added for ease of use, followed by additional layout options and layer modification functionality. At the time of writing, PDFStitcher has been downloaded over 24k times and translated into 12 languages. Additionally, a Google Docs spreadsheet containing pattern-specific parameters was created and shared by the sewing community, containing over 300 unique entries.

These numbers indicate strong support from the sewing community. Care has been taken to ensure that PDFStitcher does not violate usage agreements by keeping all processing local to the user, and to date there has been no opposition from pattern designers, with some expressing gratitude that PDFStitcher helps to make their patterns more accessible.

The most common complaint amongst users relates to Windows Defender and macOS Gatekeeper, which can flag PDFStitcher as "potentially malicious". For many users PDFStitcher is the first piece of software installed from an unofficial source, and this requires both trust and technical acumen on the part of the end user.

3.3 Future directions

PDFStitcher fills a gap in the landscape of digital sewing pattern distribution. As more designers recognize the use of projectors for sewing and begin offering purpose-designed projector files, the primary function of page imposition with PDFStitcher will become unnecessary. However, it is clear that the use of sewing patterns is evolving beyond the traditional paper format, which raises a new question: is PDF the appropriate format for digital sewing pattern distribution?

PDF preserves the visual appearance of its contents across multiple platforms and software packages, but is a difficult document to edit, containing little contextual information about its contents. While there are some promising initiatives to enhance editability of PDFs [4], sewing patterns in particular follow a semi-structured format with predictable elements, and are thus a good candidate for semantic information extraction. In addition to multiple clothing sizes commonly structured as separate OCGs, sewing patterns contain an additional subset of objects that convey information to a human reader, including:

- Annotation lines such as grainlines, fold lines, and lines indicating where to lengthen or shorten.
- Instructions such as "cut two mirrored from main fabric, cut one from interfacing".
- Optional pieces, such as collar or sleeve variations.
- Alignment notches, which may be shown as lines, filled shapes, or glyphs.
- Pattern pieces common to all sizes, such as pocket bags.

Each of these items is just another text or vector element, indistinguishable from the main pattern pieces by PDF viewing software. While parsing semantic information from academic documents and technical drawings has received considerable attention (e.g. [9, 3]), there has been little to no published work regarding parsing of home sewing patterns.

Incorporating semantic document understanding into PDFStitcher, along with an interactive graphic display of pages, would enable more advanced features such as selective display of individual pattern pieces, mirroring and replicating of individual pieces, optimizing layout, and more.

Ultimately, the sewing community would be well served by an open document format that preserves the intellectual property of the designer while allowing the sewist to modify the display of pattern pieces and perform small adjustments such as lengthening or shortening. However, for the time being, PDF is both the *de facto* standard and the best compromise between editability and cross-platform portability.

4 CONCLUSION

PDFStitcher is an open-source cross-platform GUI application that has been developed to meet the needs of projector sewing enthusiasts. It provides a set of tools to tile pages and perform operations such as trimming and rotating, as well as tools to selectively modify line properties and hide or remove OCGs from the document. PDFStitcher has been well-received in the sewing community and continues to evolve to meet the changing needs of home sewists.

REFERENCES

[1] Elizabeth Caven. 2021. Method of formatting sewing patterns for paperless use. (May 11, 2021). U.S. pat. Patent No. 11003903B2. Jo-Ann Stores LLC. Retrieved June 17, 2022 from https://patents.google.com/patent/US11003903B2/en?oq=11003903.

[2] Elizabeth Caven. 2020. Method of projecting sewing pattern pieces onto fabric. (Aug. 25, 2020). U.S. pat. Patent No. 10750810B2. Jo-Ann Stores LLC. Retrieved June 17, 2022 from https://patents.google.com/patent/US10750810B2/en?oq=10750810.

[3] Melissa Cote, Alireza Rezvanifar, and Alexandra Branzan Albu. 2020. Automatic Generation of Electrical Plan Documents from Architectural Data. In *Proceedings of the ACM Symposium on Document Engineering 2020* (DocEng '20). Association for Computing Machinery, New York, NY, USA, (Sept. 29, 2020), 1–4. ISBN: 978-1-4503-8000-3. DOI: 10.1145/3395027.3419598.

[4] Tamir Hassan. 2018. Towards a Universally Editable Portable Document Format. In *Proceedings of the ACM Symposium on Document Engineering 2018* (DocEng '18). Association for Computing Machinery, New York, NY, USA, (Aug. 28, 2018), 1–4. ISBN: 978-1-4503-5769-2. DOI: 10.1145/3209280.3229083.

[5] Karen LaBat, Carol Salusso, and Jongeun Rhee. 2007. Home sewers' satisfaction with fit of apparel patterns. *Journal of Fashion Marketing and Management*, 11, 3, (Apr. 2007), 429–440. DOI: 10.1108/13612020710763155.

[6] Addie Martindale and Ellen McKinney. 2018. Exploring the inclusion of home sewing pattern development into fashion design curriculums. *International Journal of Fashion Design, Technology and Education*, 11, 1, (Jan. 2, 2018), 104–112. DOI: 10.1080/17543266.2017.1332241.

[7] Addie Martindale and Ellen McKinney. 2020. Why Do They Sew? Women's Motivations to Sew Clothing for Themselves. *Clothing and Textiles Research Journal*, 38, 1, (Jan. 1, 2020), 32–48. DOI: 10.1177/0887302X19872552.

[8] 2019. *PDF, Version 1.7 (ISO 32000-1:2008).* (Mar. 1, 2019). Retrieved June 20, 2022 from https://www.loc.gov/preservation/digital/formats/fdd/fdd000277.shtml.

[9] Roya Rastan, Hye-Young Paik, and John Shepherd. 2019. TEXUS: A unified framework for extracting and understanding tables in PDF documents. *Information Processing & Management*, 56, 3, (May 1, 2019), 895–918. DOI: 10.1016/j.ipm.2019.01.008.