

2025 年度 芝浦工業大学 工学部 情報工学科

卒 業 論 文

CUI ベーステキストエディタ配色生成システムへの
GUI 導入とユーザビリティ評価

学籍番号 **AL22020**

氏 名 伊藤 柊生

指導教員 篠埜 功

提出日 2026 年 2 月... 日

概 要

現在、様々な項目についてカスタマイズ可能なテキストエディタが人々に広く使われており、その中でもテキストエディタの配色を自分の好みに変更するユーザーも多く存在する。配色の変更には、テキストエディタの拡張機能やインターネット上で公開されているものを利用することができるが、好みに合った配色を探すのには時間がかかる。また、自分で配色を独自に作成することも可能であるが、設定項目が非常に多い場合、自分の好みに合った配色を作成するのは時間がかかる。本研究では、ユーザーの好みを反映しつつ、視認性の高いテキストエディタの配色を簡易的に生成する手法を提案する。本手法は

目次

第1章	はじめに	1
第2章	関連研究	4
2.1	テキストエディタにおける配色の自動生成	4
2.2	web ページにおける配色の自動生成	5
2.3	インターフェースの優劣	5
2.4	ユーザビリティの評価方法	6
2.5	UI 設計	6
第3章	提案手法	8
第4章	実験	9
4.1	実験手順	9
4.2	実験結果	10
第5章	結論と今後の課題	11
	謝辞	12
	参考文献	13

第1章 はじめに

Visual Studio Code や Vim など様々な項目を設定可能な高機能テキストエディタがいたるところで多く使われている。Stack overflow annual developer survey[1]の開発環境調査によると、2024年5月の調査では、図 1.1 のように、質問に回答した開発者 54,175 人のうち 73.6%が Visual Studio Code を使用している。

これらのテキストエディタの多くは、プログラムの視認性を高めるために、字句の種類ごとに色分けを行う機能を備えており、ユーザーはこれらの色設定を自由にカスタマイズすることが可能である。テキストエディタにおける色設定の手段は大きく二つに分類できる。第一の手段は、既に作成された配色データをダウンロードして利用する方法である。Visual Studio Code においては、テーマを拡張機能としてインストールすることで導入が可能であり、Visual Studio Marketplace には、約 37,000 件に及ぶ拡張機能が登録されている。各拡張機能は一つ以上のテーマを提供しており、ユーザーはそこから好みの配色を選択する。しかしながら、これらの提供されているテーマの数が膨大であるため、希望に合致する配色を見つけ出す作業は容易ではない。第二の手段は、ユーザー自身が個別に色設定を行う方法である。Visual Studio Code においては、ワークベンチ関連の色設定項目が約 780 項目ある [2]。なお、ワークベンチの色設定においては、項目間に多くの依存関係があり、他の設定項目の色をもとに自動計算される項目も多くある。

このように、全ての設定項目を網羅的に記述することにより、自分好みの配色テーマを作成することはできるが、設定すべき項目数が多く、さらにそれぞれに対して適切な色を選定する必要があるため、作業には多大な労力を要する。一方でテキストエディタの配色に関する研究は吉越ら [3] の研究を除いて筆者らが調べ

た限り見当たらない。

吉越らの研究で推薦された手法では、ユーザがRGB形式の入力によって指定した色に基づいて、テキストエディタの配色ファイルを生成し、その後、ユーザが納得のいくまで修正を繰り返し、好みの配色を生成することができる。しかし、CUIのみを用いたツールであるため、ユーザーにとって不便であるという課題がある[3]。そこで本研究ではユーザーがより容易に好みのテキストエディタの配色の設定行えるようにするため、GUIを導入することでツールの改善を行う。

GUIの導入は、まず、吉越らのツール[3]ではCLIで行われていて不便であった文字色の選択の部分に対し、カラーピックを用いることにより行う。また、変更後の配色は吉越らのツール[3]では生成した配色をVSCodeに反映させない限りツールの使用者は見るができなかったが、変更後の配色を適用前にサンプルの文字に色をつけることで、色選択と同じ画面で確認できるようにツールの改善を行う。

改善を加えたツールの有効性を示すため、既存のツールと改善を加えたツールを被験者に使用してもらった実験を行う。ツール使用後に system usability scale[4]でアンケート評価を行った結果、....{**Todo: 実験後にうめる**}。

本論文の構成は次のようになっている。まず、2章で本研究の関連研究について述べる。次に3章では提案手法について述べる。さらに4章ではツールの比較実験について述べ、最後に5章で結論と今後の課題について述べる。

第2章 関連研究

{Todo: 次回2章以降も見る。} これまでテキストエディタにおける配色生成に関する研究は、筆者の知る限り、吉越 [3] らの研究を除き、行われていない。一方で、テキストエディタの配色に近い研究として、Web ページの配色生成システムに関する研究 [5] などがあり、視認性を高くすることを目的に配色を生成するといった手法を論じている。さらに、インターフェースの改良には、あらゆる評価手法を用い、それらを比較、評価することが重要であり、これについて様々な研究 [6, 7, 8, 4] が行われてきた。以下では、上記の内容について論じる。

2.1 テキストエディタにおける配色の自動生成

吉越 [3] **{Todo: 情報処理学会論文誌にそのうち掲載されるので、掲載後はそれで置き換える }** は、テキストエディタにおけるユーザの好みを反映した視認性の高い配色を推薦する手法を提案した。この研究では、ユーザが好みの一色を一つの文字色として指定し、その色に対し焼きなまし法を用いることで、視認性が高くなるような背景色やほかの文字色を自動生成し、visual studio code の配色設定ファイルとして出力する手法を提案した。また、ユーザが望む場合、変更したい色を指定することで、生成した配色の微調整することができる。さらに、手動で配色を作成した場合と提案した手法を用いたツールで配色を生成した場合における作成時間、配色の好み、配色の見やすさを比較する実験を行った結果、この手法の有用性が確認できた。しかしながら、視認性が非常に悪い配色が生成される場合があることと、CUI のみを用いたツールのため不便であることが、課題であると

述べている。

2.2 web ページにおける配色の自動生成

藤本ら [5] は、マルチエージェント強化学習が用い、web ページの配色生成の自動化のためのシステムを作成した。配色の視認性を高めるために、Web content accessibility guidelines[9] において推奨される、web ページの背景色と前景色とのコントラスト比の制約を利用した。また、さらなる配色の最適化のために、生成された配色に対し、ユーザが評価を行い、それらを新たにシステムにフィードバックとして与える方法を用いている。

2.3 インターフェースの優劣

Gaff[6] は様々な性格タイプの被験者を集め、著者名やタイトルの検索課題を CUI と GUI でそれぞれ行ってもらい、作業におけるやりやすさやストレスの程度を評価する実験を行った。実験結果はすべての被験者が GUI を使いやすかったと回答した一方で、6 割の被験者が CUI の利用にストレスを感じたと回答したことで GUI の有意性を示した。

また、Feizi ら [7] は、GUI と CLI どちらかのみを用いたタスクや二つを併用したタスクを用意し、17 人のインターフェースデザイナーと 15 人のソフト開発者の被験者にそれらを実行させる実験を行った。実験では複数の項目からなるユーザアンケートによる評価とタスクの成功率や平均のタスク時間を計測し、比較を行った。実験結果は GUI を用いたタスクの成功率は 100% だったのに対し、CLI を用いたタスクの成功率は 91.93% であった。また、タスクの平均時間も GUI を用いたタスクの方が短く、タスク終了後のユーザアンケートでは、画面デザインやソフトの機能性といった項目の評価平均点が高かったのに対し、「スクリプト入力のわかりやすさ」といった CLI に関する項目の評価平均点が低かった。このことから実

験で用いたソフト、Adobe Flash CL4においては、CLI よりも GUI のほうが作業を行うことに優れているインターフェースであるということが示された。

2.4 ユーザビリティの評価方法

ユーザビリティとは JIS 規格において「特定のユーザが特定の利用状況において、システム、製品又はサービスを利用する際に、効果、効率及び満足を伴って特定の目標を達成する度合い」と定義されており [10]、これを評価することは、インターフェースの改善において重要である。

Nilesen[8] は、ユーザビリティを評価するための方法として、ユーザビリティの専門家が評価を行う、ユーザビリティインスペクションを提言した。その主な手法には、専門家が 10 のユーザビリティ原則に基づいて判断するヒューリスティック評価や、ユーザーの問題解決プロセスをシミュレートし、各ステップでのユーザーの目標と記憶内容が次の正しいアクションにつながるかどうかを確認するコグニティブ・ウォークスルーなどを含む、6 つの手法がある。

さらにほかのユーザビリティ評価方法として、Brooke[4] が提言した、system usability scale がある。system usability scale はユーザビリティの専門家ではなく、システムのユーザーが図 2.1 のシステムの仕様に関する 10 項目の質問に対して、5 段階のリッカート尺度で回答し、それらを決まったルールに従って計算を行い、算出されたスコアを評価とする。

2.5 UI 設計

図 2.1: system usability scale における 10 項目の質問

第3章 提案手法

本研究では JIS 規格に基づいた、CIE 1976 $L^*a^*b^*$ 色空間 [11] および RGB 色空間を用いる。またこの RGB 色空間というのは sRGB 空間 [12] を示すものとする。

第4章 実験

本章では、3章で提案した GUI 導入後ツールと、既存の CUI ベースツールを用いて、Visual Studio Code を使用したことがある人々を対象に、それぞれのツールで場合にユーザビリティにどのような違いがでるのか検証する実験について記述する。

4.1 実験手順

はじめに、被験者は既存の CUI ベースツールを用いて、テキストエディタの配色作成を行う。このツールはユーザがコマンドライン引数を用いて、カラーコードで好みの色を入力することによって、自動で色に基づいて配色を生成し、設定ファイルを作成する。その後、ユーザは設定ファイルを Visual Studio Code で反映させ、さらに調整したい色をコマンドラインでコマンドを用いて選択し、満足のいく配色を作成する。これを制限時間 60 分で行い、時間内に満足のいく配色を生成できなかった場合でも終了とする。時間計測は、最初の色を指定するために css ファイルを用いて、カラーピッカーを開いた時点から最終的な配色が Visual Studio Code に反映されるまでの時間を計測する。その後被験者は system usability scale の 10 項目の質問に回答する。

次に、改良後の GUI 導入ツールを用いて、テキストエディタの配色作成を行う。このツールは **{Todo: 具体的な実装が決まってから埋める }** これを制限時間 60 分で行い、時間内に満足のいく配色を生成できなかった場合でも終了とする。時間計測は、最初の色を指定するためにカラーピッカーを開いた時点から最終的な

配色が Visual Studio Code に反映されるまでの時間を計測する。その後被験者は system usability scale の 10 項目の質問に回答する。

4.2 実験結果

それぞれのツールにおいて、配色生成にかかった平均時間と system usability scale のスコアは以下ようになった。

第5章 結論と今後の課題

謝辭

参考文献

- [1] Stack Overflow. Stack overflow annual developer survey. <https://survey.stackoverflow.co/>. 2025 年 04 月 28 日参照.
- [2] Microsoft. Theme Color Reference — Visual Studio Code Extension API. <https://code.visualstudio.com/api/references/theme-color>. 2025 年 05 月 20 日参照.
- [3] 吉越丈. 好みを反映した視認性の高いテキストエディタの配色を推薦する手法の提案および実装. 芝浦工業大学工学部情報工学科 2024 年度卒業論文, 2025.
- [4] Brooke J. Sus: A “quick and dirty” usability scale. In *Usability evaluation in industry*, pp. 189–194, 1996.
- [5] 藤本祥, 市川嘉裕, 山口智浩. Web ページの配色のためのインタラクティブな推薦システムの試作. 情報処理学会第 83 回全国大会講演論文集, pp. 323–324, 2021.
- [6] Matthias Rauterberg. An empirical comparison of menu-selection (CUI) and desktop (GUI) computer programs carried out by beginners and experts. *Behaviour & Information Technology*, Vol. 11, No. 4, pp. 227–236, 1992.
- [7] Andisheh Feizi and Chui Yin Wong. Usability of user interface styles for learning a graphical software application. In *2012 International Conference on Computer & Information Science (ICCIS)*, Vol. 2, pp. 1089–1094. IEEE, 2012.

- [8] Jakob Nielsen. Usability inspection methods. In *Conference companion on Human factors in computing systems*, pp. 413–414, 1994.
- [9] WAIC. Understanding WCAG 2.1. <https://waic.jp/translations/WCAG21/Understanding/use-of-color.html>. 2025 年 05 月 20 日参照.
- [10] 日本産業標準調査会. 人間工学－人とシステムとのインタラクション－ユーザビリティの定義及び概念. JIS Z 8521:2020. 2020.
- [11] 日本産業標準調査会. 測色-第4部:CIE 1976 L*a*b*色空間. JIS Z 8781-4:2013. 2013.
- [12] 日本産業標準調査会. 測色－第6部: CIEDE2000 色差式. JIS Z 8781-6:2017. 2017.