

組込みシステム開発のデバッグにおける視線と熟練度の関係分析

馬場 建
Takeru BABA

1 はじめに

IoT 市場規模の拡大に伴い、IoT の分野の技術者の需要が増加すると予想されている¹⁾。IoT 分野の開発では、組込みシステム開発の要素に加え、ネットワークの知識が必須であり、技術の習得が容易ではない。組込みシステムの要素であるハードウェア (以下、HW) とソフトウェア (以下、SW) に加え、IoT ではネットワークの知識が必要となる。IoT の基礎である組込みシステム開発の効率化は、今後のソフトウェア工学の課題となると考えられる。

組込みシステム開発は、一般的な SW 開発に加えて HW の要素が増える。組込みシステム開発におけるシステムが正常に動かない原因、すなわち誤りの箇所は、SW だけでなく、マイコンや回路などの HW にも発生しうる。多くの場合、HW の誤りは SW の誤りと見分けがつかないため、システムの誤り箇所を推定し、修正することは初学者にとって容易ではない。

多くの要素を同時に考慮しながら行われる組込みシステム開発のデバッグにおいて、熟練度による誤りへの対応の差があると考えられる。熟練度の差、すなわちコツを明確にし、初学者に教示することでデバッグ効率が向上すると予測できる。したがって本研究では、熟練者のコツを分析して初学者へコツの教示を行うことで、組込みシステム開発における初学者のデバッグ効率向上を目的とする。

2 視線情報による熟練度の差の分析

2.1 視線の動きと分析手法

人間の視線の動きは、高速移動と停留の動作に分けられる。停留の動作は、注目物体の移動速度が低速なときや、静止物体を見る際に起こり、視覚認知の多くが視線の停留中に行われている。

視線の停留点の座標に関する分析手法は、統計的分析、時系列分析、認知プロセスの推定の三種類ある。統計的分析は注目物体に関する時間的な分布を、時系列分析は注目の順序関係を調べる方法である。そして認知プロセスの推定は、PC の作業ログと視線の遷移を関連付けることで、人の認識過程を推定する方法である。

2.2 視線遷移のモデル化と利用

一般に、視線には無意識的な暗黙知が反映されるため、視線分析により暗黙知が定量的に表せるとされている。Paul は、ソースコードリーディングにおけるコツを視線から分析し、ソースコードの要素の意味に関する遷移を隠れマルコフモデルとして表した²⁾。

視線の分析によって得られたコツを言語化して作業者に教示することで、作業効率が上がることが一般的に知られ

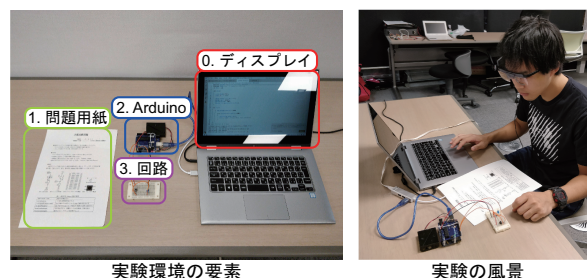


Fig.1 実験環境の要素と実験の風景

ている。應治らは、ソースコードレビューにおける熟練度の差を視線移動から計測し、設計書やソースコードの読み方を作業者に教示することでレビュー効率が変化するかを検証した³⁾。実験の結果、教示ありのグループは誤りの発見速度が上昇し、誤りの発見率も向上したと報告している。

3 組込みシステム開発に関する視線分析実験

3.1 実験概要

本実験の目的は、組込みシステム開発において、視線情報から数学的モデルを生成することで、熟練度の差の要因を顕在化することである。差を分析するために被験者に課題を与え、視線情報とデバッグの操作ログを記録する。

3.2 実験条件

被験者は、組込みシステム開発経験がある大学生の六名である。組込みシステム開発経験が一年以上である二名を「熟練者」、一年未満である四名を「初学者」と定義する。

実験は Fig. 1 に示す環境で実施した。実験環境の構成に用いた組込みシステムの要素を下記に述べる。

0. ディスプレイ (ソースコードと文字出力)
1. 問題用紙 (回路や関数、コンパイル方法などを記述)
2. Arduino (電圧を制御するマイコン)
3. 回路 (デジタル入出力に関する回路)

被験者に与える課題は、スイッチの押下時に一定時間間隔で LED が点滅し、押下時以外は LED が消灯するシステムの構築である。

3.3 実験手法

被験者には、ソースコード側と回路側にそれぞれ 2 つの誤りが埋め込まれている状態のシステムを与える。ソースコード側には、動作の待ち時間に関する誤りと、回路の状態を正しく反映できるかを問う誤りがある。回路側には、ブレッドボードの使用法と、電子部品の知識が必要な誤りがある。実験開始時にはすべての被験者に対して、システムの完成形を見せながら課題内容やコンパイル方法の説明を行った。また、実験に用いる開発言語が未経験の被験

Table.1 平均回答時間と物体の平均注目時間の割合

	時間 [分]	p0	p1	p2	p3
熟練者	13	0.66	0.17	0.03	0.14
初学者	32	0.59	0.20	0.05	0.16

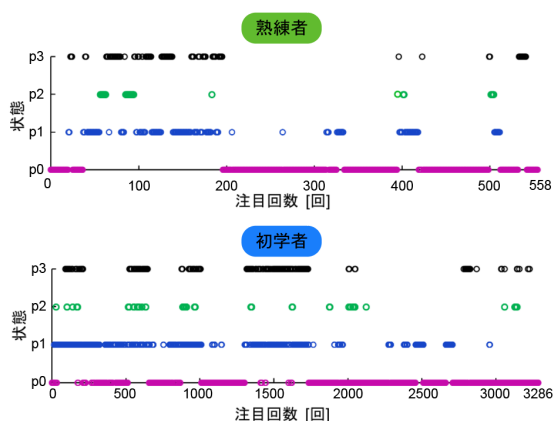


Fig.2 ある熟練者と初学者の注目物体の時間的推移

者や、回路図による回路の実装経験がない被験者に対しては、関数や回路図の説明を簡潔に行った。説明の後、被験者はシステムのデバッグ作業を行った。実験中に取得する情報は、アイトラッカーから得られる停留点の座標に関する視線情報、コンパイル結果、1秒ごとのソースコード編集履歴の三種類である。

3.4 実験結果と考察

被験者実験で取得した情報から、熟練度の差を起す要因を分析した。熟練者と初学者の修正にかかった平均時間と、各物体を注目していた時間の割合を Table. 1 に示す。p0～p3 はそれぞれディスプレイ、問題用紙、Arduino、回路のいずれかを注目している状態である。熟練者は初学者と比較して短時間で誤りを修正しており、熟練度の差がデバッグにかかる時間に影響していることがわかる。物体の注目時間の割合は被験者によってばらつきが大きく、有意な差は見られなかった。

次に、実験開始から終了までの視線の停留点ごとの注目物体の推移を分析した。ある熟練者と初学者の注目物体の推移を Fig. 2 に示す。熟練者は序盤に回路を中心に注目する傾向があり、初学者は序盤に回路とディスプレイ間の注目の推移を繰り返す傾向があるように見える。

熟練度により傾向が違ったのは、熟練者は先に回路を修正するという方針があったのに対し、初学者は誤り箇所を推測できず、修正の方針が定まらない状態に陥ったことが原因だと考えられる。したがって、初学者に修正方針を教示することで、デバッグ効率が向上する可能性がある。

熟練者がどのような方針でデバッグを行うかを調べるために、実験結果を三分割して分析することで、時系列的なデバッグ箇所の違いがあるか検証した。熟練者と初学者の時間分布を三つの段階で調べると、中盤のディスプレイに関する時間分布に大きな差があった。

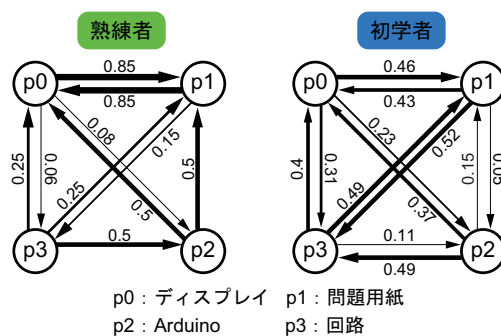


Fig.3 実験の中盤における注目物体のマルコフモデル

実験結果を分割することにより時間分布に違いが現れたため、各段階における注目物体の遷移確率をモデル化し、傾向が異なるかを詳細に調べた。熟練者と初学者の実験の中盤における、注目物体のマルコフモデルを Fig. 3 に示す。矢印の太さは、遷移確率の高さを視覚的に表したものであり、矢印が太いほど確率が高い。実験時間全体をまとめて分析した場合、熟練度による遷移確率の有意な差が見られなかった。しかし、実験結果を分割して分析した場合、Fig. 3 の回路へ向かう遷移のように、各段階において一部の遷移確率で有意な差があった。このことから、実験時間を正しく分割して分析することで、熟練者の誤り修正の方針を明らかにできると考えられる。

4 今後の研究方針

現段階の分析結果では、熟練度による明確な違いを発見できていない。したがって、今後は次の三つのことを行う。一つ目は、動作の傾向ごとに分割して実験結果を再分析することである。実験結果を分割することで遷移確率に有意な差が見られたため、実験時間の傾向に基づく分割方法を用いて再分析をする必要がある。二つ目は、本実験で判明した熟練度の差の要因を言語化することである。熟練者の持つコツを、初学者に簡潔に伝えられるようにする必要がある。三つ目は、言語化したコツを初学者に教示し、デバッグ効率が向上するかを検証することである。デバッグ効率の向上があった場合、初学者へのコツの教示が組み込みシステム開発の学習支援に有効であると考えられる。

参考文献

- 1) 野村総合研究所, IoT 市場の拡大と、日本における IoT 活用のありかた, https://www.nri.com/-/media/Corporate/jp/Files/PDF/knowledge/publication/m_review/2017/nmr38/nmr38-1.pdf?la=ja-JP, 参照 Oct.20, 2019.
- 2) Paul A. Orlov, Primary investigation of applying Hidden Markov Models for eye movements in source code reading, Eye Movements in Program Education, pp.18-20(2015).
- 3) 應治 沙織, 上野 秀剛, コードレビュー時の読み方教示によるレビュー効果の変化, 情報処理学会研究報告書, Vol.2014-SE-185, pp.1-8(2014).