



図-2 ブリンクするオレンジの円でクリックアイテム候補を提示

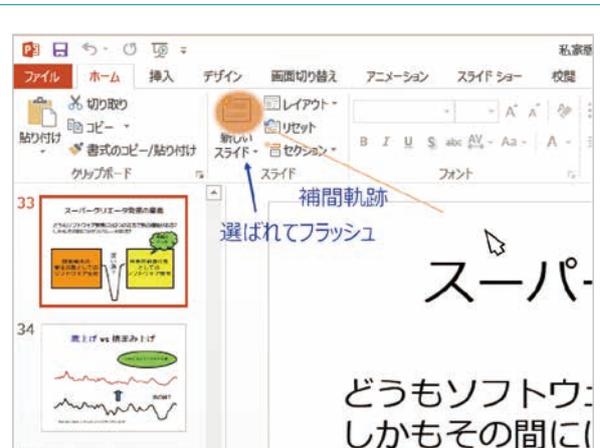


図-3 飛んだカーソルの軌跡が一瞬見えて、自動クリック

白いソフトをいくつか書いた。

そのコンサルティングファームでの教訓(?)が「スライド 2,000 枚書いて初めて一人前だ!」だった。それはともかく、ビジネスソフトのUIが少しでも悪いと、多数のユーザがいる会社全体ではそのために無駄になった時間の費用が億円単位になる。

安野君はこう考えた。人によって使い方が異なるので、万人のための最適化はそもそも不可能。だから、個人の操作ログに応じパーソナライズされたUIを提示すればいい。しかし、UIを変え続けながら最適解を探すことはできない。だから、元のUIは変えず、既存のUIの上に新しいレイヤを作ればいい。

提案は、一般のビジネスソフトのクリック位置(とそれに伴うカーソル移動)を機械学習により予測できるようにし、それをユーザに適切に提示して、マウスを握ったままショートカット的なことができるシステムだった。最終的にはパワーポイントに対象を絞って、成果を尖鋭化させた。これを可能にする仕組みを Lightning UI と命名した。

動きのあるUIを紙上で紹介するのは辛い。その仕組みを簡単に紹介しよう。パーソナライズのために、ユーザのクリックを検知し、どのアイテムが押されたかを画像データから取得すると同時に、ユーザ名、日時、そのときアクティブだったソフト、スクリーンショット画像等を学習用の行動ログとして蓄積していく。

次のクリック位置は、マウスの少しの動きからその方向の先の範囲にあるアイテムを抽出し、行動ログが

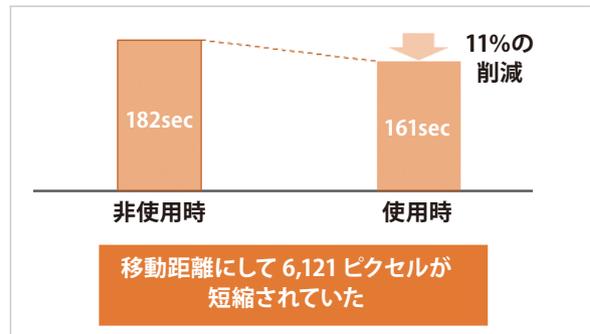


図-4 Lightning UIによる操作の短縮効果

ら最も高く評価されたアイテムを予測する。選ばれたアイテムをブリンクするオレンジの円で囲む。ユーザはそれでOKであれば、マウスをそこまで動かさずに、あらかじめ指定したキーボードアクションで実際にクリックする(図-2)。

デモを見ると、マウスカーソルが少ししか動いていないのに、アイテムまでカーソルがワープしたような軌跡が一瞬見えて、クリック表示がなされる。カーソル飛ばしゲームをやっているように見える(図-3)。

ある実験では11%の速度向上、6,121ピクセルの移動距離が短縮されていたとのこと(図-4)。劇的な数値ではないが、これが積み重ねれば巨額の生産性向上につながるのだろう。

意外だったのは、深層学習ではなく、遷移行列による予測が最もの中率が高かったこと。適材適所なのだろう。

プロジェクトは試行錯誤の連続だった。カーソルの周りにボタンを配置したり、多機能マウスを使ったり、



図-7 PC上で見える心温計のダッシュボード

実は図-6のiPhoneの画面の下のほうに、石丸君の修士論文前の心温変化(左)と、未踏成果報告会前の心温変化(右)が並べて表示されている。修論発表では、その直前に心温が上がりっぱなしになっていることが分かる。しかし、その後の未踏成果報告会の前は、自分で心温をコントロールする努力をしたとのこと。こうやって少し長い時系列で見ると、心温計の意義がよく分かってくる。心温は、相対値の変化がそれぞれのユーザにとって意味があるのだろう。また、心温計のユーザは心温計をうまく活用して、心温を自分でコントロールできるようになるかもしれない。

心温の表示はiPhoneだけではなく、PCでも可能である。心温の根拠となる各活動量は、iPhoneでも同等の表示が可能だが、PCではより見やすい。これをダッシュボードと呼んでいる(図-7)。ユーザが心温を安定させるための行動指針も得られそうだ。

新たなセンサをアドオンできるように設計されているのも素晴らしい。つまり、石丸君は心温計設計の基盤を開発したのだ。また、センサを全部使わなくても心温が測れるようになっている。

こういうシステムを開発しても、医学界から拒否されては意味がない。2人の医師からアドバイスをもらって、社会展開への可能性を塞がない、あるいは可能性を広げる努力もした。

石丸君が心温計を着想したのは、自らが一時的な鬱状態になって苦しんだからである。モチベーションが明快なので鬼に金棒である。担当の首藤PMが最

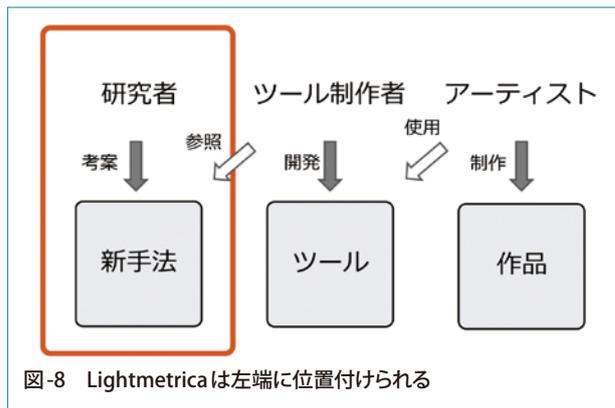


図-8 Lightmetricaは左端に位置付けられる

終評価で述べているとおり「石丸君はデータに基づく心の診断の新しい未来を拓きつつある」。

(首藤一幸 PM 担当)

おおつ ひさなり
■ 大津 久平

大域照明計算手法開発のためのレンダリングフレームワーク^{☆4}

これはコアテクのプロジェクトだ。目の前に直接的な産業応用が見えているわけではない。だが、未踏は多様性が命である。1年後が勝負のプロジェクトもあれば、5年後や10年後に勝負をするプロジェクトもある。後者の芽を摘んでしまっては日本が廃る。

大津君が開発したのは、レンダリング技術の参照基準となり得るフレームワーク Lightmetrica である。レンダリング研究がいまでも盛んなのは、多様なシーンに対応できる高速・高効率な手法のニーズと表現可能な幅を広げるニーズがあるからである。映画製作で使われるアートのCGレンダラは新規レンダラ開発の参照基準やプラットフォームにはなり得ない。

新手法を素早く従来手法と比較したり、検証したりするには、使いやすく拡張可能性に優れたレンダリングフレームワークが必要だ。Lightmetricaは、C++で書かれており、Windows, Mac OS, Linuxで動作する。研究者からアーティストへ、新手法の開発がどう伝わるかを図-8に示した。Lightmetricaのターゲットは左端の赤枠の部分である。

^{☆4} <http://lightmetrica.org/>



図-9 3種類のマテリアルの表現



図-10 Lightmetricaのシンボルと言える不思議な物体

Lightmetrica は研究者のためのツールだが、作り方が徹底している。コア部分は小さく、必要な多くの構成要素をプラグイン可能にした。また、同一の手法をシンプルに実装可能にした。たとえば、双方向パストレーシングという基本手法の実装は、有名な Mitsuba では約 2,000 行かかるが、Lightmetrica では 700 行で済む。研究者は自分の開発した手法を、短い行数でプラグインとして実装すれば、簡単に比較検討できる。また、多種類のテストで検証できるので、バグの発生を抑制できる。要するに拡張可能性(レンダリング手法、交差判定、マテリアル、光源モデル、センサモデル、テクスチャ、メッシュ等)、拡張の容易さ、テストによる検証で、Lightmetrica は既存システムを凌駕している。

Lightmetrica はもちろんレンダラとして機能するので、その作品を 2 つ紹介しておこう(図-9, 10)。大域照明なので、光源からの光の反射光、その光の反射光、と連鎖して、非常に複雑な計算が必要になる。

図-9 はプラグインとして実装された 3 種類のマテリアルである。左から光沢反射、拡散反射、鏡面反射。これらのマテリアルの実装はほかの手法の実装に影響を与えない。つまり、ほかは同一の実装のままでよい。

図-10 はカラーではないが、Lightmetrica の Web ページのトップを飾る CG である。大津君の思いがこもっているのであろう。形状を見ると相当複雑な計算が必要と思われる。

こういうシステムを作っても、ドキュメントが弱くて世界に拡散しない事例をよく見かけるが、大津君はまず英語でドキュメント整備をした。日本語で書くより英語で書くほうが楽なのだそうだ。帰国子女ではない。こういう若者が出てくることこそ重要だと思う。もちろん、日本語のドキュメントも整備されている。

成果報告会で、大津君はどこかしら軽々と話をしていたので、C++ で一体何行のプログラムを書いたのかという、かなり下世話な質問をしたら、「5 万行ぐらいでしょう」とこれまた軽々と答えてくれた。

実は大津君、私の古巣の東大情報理工学系研究科の創造情報学専攻の学生で、指導教員は蜂須賀恵也講師である。蜂須賀君は 2002 年、当時私が始めた「未踏ユース」のスーパークリエイターである。米国での長い活躍から戻ってきた。その蜂須賀君、大津君に「そういうツール作りもいいけれど、本当の研究もしてね」と言っていたと聞いた。どんな形であれ、人々は連綿とつながっていくものだとな納得した次第である。

なお、Lightmetrica はオープンソースソフトウェアとして、すでに脚注にある Web ページで一般公開中である。
(後藤真孝 PM 担当)

■ たけうち りと やまなか おさむ
竹内 理人, 山中 治

UMATracker : 動物の集団行動映像から個体の動きを識別する^{☆5}

日本動物行動学会では群れの行動を解析する研究が行われている。単に科学的興味というだけでなく、化学物質入りのエサと群れの行動の因果関係を調べ

☆5 <http://ymnk13.github.io/UMATracker/>

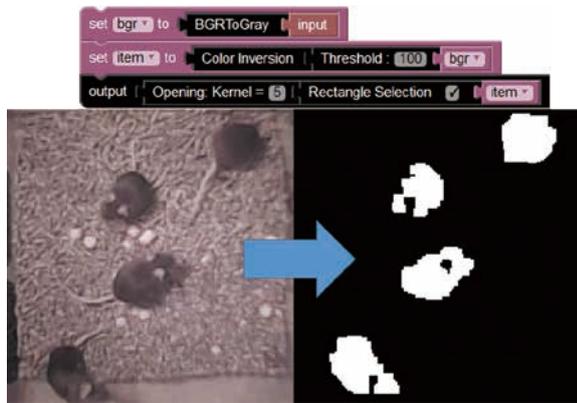


図-11 ネズミの動画の前処理

ることによって、医薬品の開発にもつながる。

いろいろな動物が対象となるが、たとえば水槽の中で泳ぐ多数のメダカの行動を観察したいとしよう。従来、多くの研究者は映像の静止画を見ながら、1匹1匹のメダカをマウスでクリックしながら追跡していた！これを「手打ち」と呼ぶ。2分の動画の手打ちに2時間もかかる。これを自動化するソフトも存在するが、前処理、分かりにくさ、失敗の回復などいろいろな面で不満が多く、研究者には敬遠されていた。

竹内君と山中君が開発したUMATrackerはこの2時間を、鼻歌混じりの数分に短縮してしまう。

まず個体追跡をしやすくするための前処理をGoogleのBlocklyというビジュアルプログラミングを用いて行う。画像フィルタをビジュアルに組み合わせていき、WYSIWYGで完成させる(図-11)。これは自閉症ネズミの研究での例である。

次に実際の個体追跡を行うが、既存ソフトよりずっと簡単にパラメータ調整ができるようにし、研究者の負担を下げた。たとえば、図-12は34匹のメダカの個体追跡結果である。個体追跡アルゴリズム自体はプラグイン可能なので、対象とする動物に特化したアルゴリズム導入が容易である。この機能がUMATrackerの普及に大いに貢献した。

アルゴリズムを精緻にしても、たとえば2匹のメダカが微妙に交差して泳いだ場合など、追跡ミスはどうしても起こる。これを簡単に修正できないと実用にならない。UMATrackerはGUIで簡単にミスを修正できるようにした。個体追跡結果の画面の入れ替わった



図-12 水槽の中のメダカの個体追跡結果

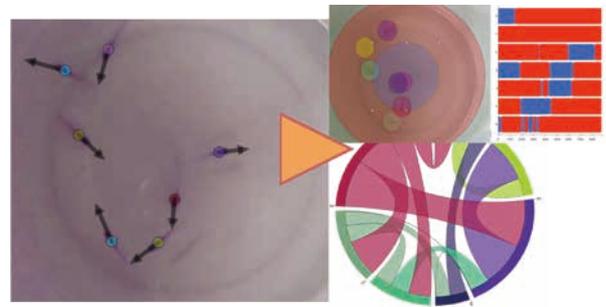


図-13 メダカの居場所や個体間インタラクションの解析

2匹付近を拡大して、Swapボタンを押すだけである。

手打ちであれ、UMATrackerであれ、得られた個体追跡結果を解析して可視化するのも大変である。研究者へのヒアリングの結果、これを知った2人はそのための便利ツールも開発した(図-13)。

タイトルの脚註に示したURLにアクセスすると、いろいろな動画を見ることができる。

2人は広島大学出身で、竹内君は大学院から東京工業大学に進んだ。2人は遠隔協力で開発を進めたが、UMATrackerを多くの研究者に使ってもらうための「営業活動」を関東・関西で分担して行った！これはシステムに必要な機能を聞き出すヒアリングを兼ねていた。それだけではなく、日本動物心理学会の大会に参加し、ポスター展示の場で、個体追跡をやりたい研究者18名にアタックした。まさに足で稼ぐ「行動」の2人である。

こうしてUMATrackerは、2016年3月の時点で、琉球大学、長崎大学、広島大学、岡山大学、同志社

大学、基礎生物研究所、専修大学、慶應義塾大学、東京大学、日本女子大学など12機関で利用されており、これを使った論文も出始めている。研究だけではなく、大学や高校での教育にも利用され始めた。未踏プロジェクト期間中に、足で稼いでこれだけのユーザを獲得し、役立った例は珍しい。獲得したユーザの多くは、それまで手打ちで頑張っていたか、既存類似ソフトの利用を断念したユーザだった。なお、UMATrackerはGithubで一般公開中である(脚注につけたURLを参照)。

メダカ以外では、マウス、キングヨ、ヒト(!)、コウモリ、アリ(山中君はアリの行動研究が本業)、ラット、イカ(半透明なので前処理が重要)、タイ、スズキ、ハト、イルカなど多くの動物を対象にUMATrackerは活躍中である。

研究者に特化した、ニッチなソフトであるが、これから世界でも広く使われて、スタンダードになる可能性は高い。ヒトも対象になるので、ヒトの集団行動の解析も可能になるだろう。マーケティングなどへの応用も見えてくる。(後藤真孝 PM 担当)

つちや ゆういちろう

■ 土屋 祐一郎

Nano Deep : 手乗り人工知能カメラ

Nano DeepのDeepは深層学習を意味している。現在深層学習が大きなブームになっている。特に画像認識では驚くべき性能を示している。しかし、ほとんどの場合、大規模な計算資源が使える環境でのみ実用的な時間で応答できるレベルである。また、IoTもブームの様相を呈している。たとえば、インターネットの端々にカメラを配置して面白いことができるようになってきた。しかし、膨大な数な端末から生画像を送ってセンターで処理するのでは、ネットワークもセンターも破裂する。末端である程度高度な処理をする、いわゆるエッジコンピューティングが必要になる。

Nano Deepは掌サイズのカメラデバイスの上で深層学習をしてしまう。大きさ、消費電力、コスト面で、深層学習でよく使われるGPUは利用できない。土屋君は深層学習に特化したFPGAを作り、GPUの

3つの問題をすべて解決しようと目論んだ。

深層学習では大量の足し算・掛け算を並列実行しなければならない。GPUはこれを得意とするが、深層学習に限って言えば、GPUは過剰に高機能である。FPGAでこれを代替すれば大きさ、消費電力、コストの問題を解決できるのではないかと。最初の大きな決断は、浮動小数点数ではなく、固定小数点数を使うこと。これで回路が簡単になり、高速になる。ただし、小数部のビット長が固定なので、一定の誤差が必ず生じる。それにもかかわらず、土屋君は32ビットではなく、16ビットの固定小数点数を使うという決断をした。シミュレーションの結果、それでもほとんど性能低下しないことが確認されたからである。こうして、FPGAでのストリーム・並列実行の余裕が増えた。さらにメモリとFPGAの間のI/O最適化も行った。カメラは直接FPGAにつながる。

使ったFPGAはXilinx社のZynq-7000である。これを使うにあたって、同期のsigboostの青木君が先達だったのでいろいろ教わったという。未踏同期たちのネットワーク効果である。Zynq-7000にはFPGAのほかに667MHzのARM CPUが備わっている。ARMではUbuntuの上でPythonライブラリが走っており、画像認識、カメラ操作、カメラ入力認識などを司る。なんだかRaspberry Piっぽい、とは土屋君の弁。それくらいお気楽に使えるということだろう。同一ベンチマークでARMとFPGAの比較をしたところ、FPGAが100~180倍高速になった。Nano Deepを写真3に示す。また、そのアーキテクチャを図-14に示す。ARMを活用して、スタンドアロンで開発ができる環境も用意している。

図-15は動画を撮りながら、streetcar, appleをリアルタイムで認識した結果である。実際に動いているところを見ると、さくさくと反応している。認識分類された画像は保存され、検索することが可能である。

Nano Deepをどう使うか、どう展開するか、考えないといけないことは多い。いろいろ考えた挙げ句、成果報告会で土屋君が見せてくれたデモは、「自分の見たものを客観的に教えてくれるライフログ」で、忘れ物防止や高齢者などの注意力診断に使えるというもの。

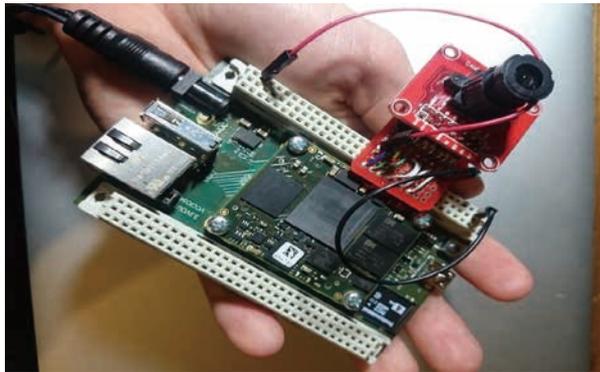


写真3 掌サイズのNano Deep

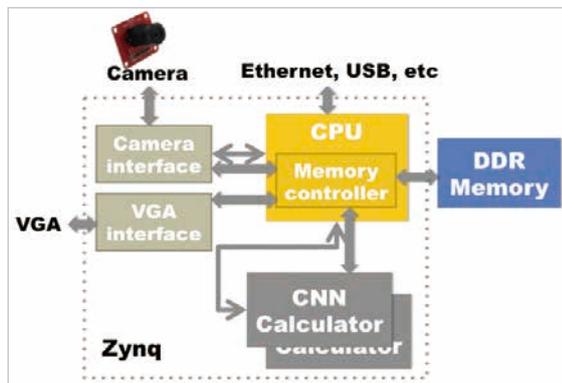


図-14 Nano Deepのアーキテクチャ

まだまだキラーアプリというには至らなさそうだ。

ウェアラブルカメラ、ドライブレコーダ、監視カメラ、ドローン搭載など、IoTに絡むようないろんなアイデアが湧いてくるのだが、それぞれに先達がいる、それを凌駕する努力が必要になりそう。しかし、それでも、これだけコンパクトな人工知能カメラなのだから、必ずドンピシャのキラーアプリが見つかるはずである。

土屋君は、Nano Deepをデベロッパ向けのプラットフォームとしてブラッシュアップした上で、Kickstarterなどのクラウドファンディングを狙っている。そういえば、まだNano DeepのWebサイトはない。満を持しているのだろう。（石黒浩 PM 担当）

寺本 大輝

HackforPlay : ゲームをハックしてプログラミングを好きになる☆6

子供にプログラミングを教えることがブーム化して

streetcar



apple



road



図-15 保存された認識結果をまとめて表示

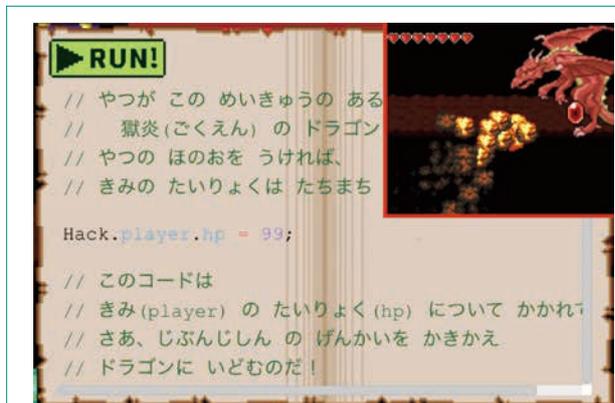


図-16 右上のドラゴンを倒すヒントになる魔道書

いるが、子供にプログラミングを教えるときに、まず考えないといけないことは、プログラミングを嫌いにさせないことである。もちろん、これはプログラミングに限ったことではない。

寺本君たち（谷口諒君との共同プロジェクト）の元々のアイデアは分かりやすい。ゲームが嫌いな子供はいない。ドラゴンと戦わないとクリアできないロールプレイングゲーム（RPG）があるが、どうやっても勝てない。しかし、そこに落ちている魔道書を開くとこのゲームのソースプログラム（JavaScript）が書いてある。プレイヤーのヒットポイントに99が代入されている（図-16）。勝てるわけがない。そこで、魔道書（それ自体がエディタになっている）の中の代入文の右辺を大きな999999999に書き換えてしまう！

☆6 <https://hackforplay.xyz/>



図-19 CSSで見かけがこんなに変わる

ないとう たかお
■ 内藤 剛生

Eyecatch: Web UI 更新の検証を効率化する☆7

UI を含んだ Web サービスを更新したときの検証はいまだに人海戦術である。サービスリリースごとにUIが更新されることも多々ある。内藤君の Eyecatch は Web サービス開発における UI 検証の継続的なインテグレーション (CI) サービスである。

内藤君は、東京理科大学卒業後、エストニアの首都にあるタリン工科大学に留学したという経歴の持ち主だが、実は Web サービス開発者としての経験が非常に豊富である。このプロジェクトはタリン工科大学を休学し、日本に戻って挑戦した。

プログラムの更新に関してはいろいろなテスト手法があり、ある程度自動化されている。スタイルシートで見た目やUIが規定された Web ページのUIでは、明確な入力と出力を定義できないので、プログラム検証の技法が使えない。結局、開発者が目視して検証するしかなかった。図-19 は、同じ HTML なのに style.css を変えたらこんな見かけが違うという例である。

UI 検証の既存 CI ツールはあるが、大量のスクリンショットを自動生成してくれるだけで、その先は、目視検査の世界である。自動化のご利益がほとんどない。変化のあったところを自動的に検出してくれないと意味がない。

Eyecatch は自動ですべてのページを確認し、変

☆7 <https://eyecatch.io/>

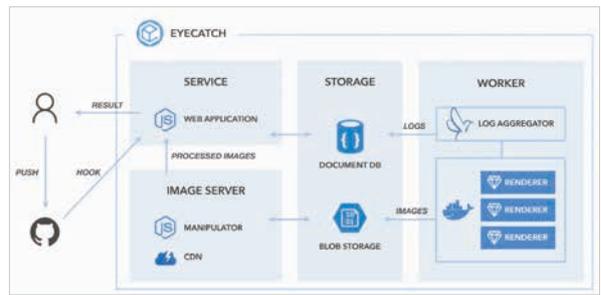


図-20 Eyecatch のバックエンドの様子



図-21 変化部分のハイライト表示 (逆にほかの部分は暗く表示)

更前と変更後で変化がないかどうかを自動検出する (図-20)。そして変化のあったところだけをハイライト提示する (図-21)。この自動検出は、Github リポジトリの変更がトリガとなって自動起動される。このビルドは隔離された環境で行うので、ほかに影響を及ぼさない。このためデータベースアプリのインストールが必要だったり、事前にコンパイルが必要だったりしても Eyecatch が使える。

UI の検証に使えるようにするため、ユーザの動作の多く、たとえばテキストフォームへの文字入力、ボタンのクリックなどを Eyecatch に代行させることができる。だから、ユーザ認証が必要なページでも検証可能である。

こう見てくると順風満帆だが、ヘッダやフッタを変えたら全ページに変化があったと提示してきて問題が出てきた。そのため、Eyecatch は画像解析により、変化の内容自体を解析し、同様の変化の繰り返しと判定した場合は、それらを1つの変化としてまとめ上げる。全ページに変化があったと提示されても「仕様上」は間違っていない。しかし「機能上」は間違っている。ページに現在時刻が表示されることもあ

るが、透明化処理により変化ではないと対処することができた。

Eyecatch が素晴らしいのは、これまでのテスト作業に求められていたテストコードの記述作業をなくし、開発者が開発を進めていく上で発生するバージョン変化によって自動的に検証が起動されるところにある。だから、導入ハードルが低く、プログラミング経験のないデザイナーでも使いこなせる。

Eyecatch の動作は HTML+CSS で作られた UI に限定されるが、画像解析やサービスアーキテクチャ自体はほかの対象にも使える汎用性がある。今後はそのあたりを充実させていこうと内藤君は考えている。

プロジェクト期間中、藤井 PM の紹介でいろいろな IT 系企業でヒアリングをすることができた。そこで得られた「これは確実な需要がある」という確信が、プロジェクトを推進することになったと思われる。

藤井 PM は最終評価報告で「本プロジェクト期間中に同じ領域の課題を解決する類似サービスが海外国内それぞれで発表されたが、いずれも Eyecatch に遠く及ばないことも追記しておきたい」と明言している。

未踏期間終了後（2016年6月現在）、Eyecatch は closed beta になっており、近々ビジネス展開に進みそうである。こういうすぐに役立つ成果で内藤君がしっかりビジネスができることを期待したい。

（藤井 PM 担当）

惜しくもスーパークリエイター認定を受けなかったものの、未踏クリエイターは優秀である。例によって私の印象に残った何人かを紹介したい。

佐藤邦彦君が開発した少年野球のピッチャーのフォームの指導システム fitform は、タブレット1つでお父さんコーチが使えるシステムである。タブレットで子供の投球動作を撮影し、簡単な UI でキー画面で骨格を指示すれば、肘や足などいろいろな角度を自動で追尾し、適正なフォームからのずれを検出する。それに基づいてアドバイスをする。コーチング技術に関する知財の問題をクリアすれば、世の中に広まりそうだ。佐藤君は以前の未踏ユースで恒例になっていた、川合賞(未踏 OB の川合秀実君が独自の観点で顕彰す



図-22 今期のクリエイターのユニークなロゴや命名

る賞)の復活第1期の受賞者となった。

和家尚希君、伏見遼平君、鈴木良平君、宗像悠里君の4人組は、音で自分の周囲を「見る」というユニークなシステム Sight を追求した。いわく「知覚旅行」のための視覚代行デバイスである。映像から音への変換なので、なかなか目で見えるようにはならなかったものの、後ろにも目があるようにする手助けなど、今後のいろいろな可能性を感じさせるものがある。

村岡眞伍君の候補提案型作曲支援ソフト OtoComplete は AutoComplete のシャレだが、その名の通り、演奏に対してリアルタイムで次のコード進行をビジュアルに提案してくれる。提案時から中身がそれほど大幅に充実しなかったのは残念だが、村岡君のプレゼンの進化は、成果報告会ではベストプレゼンと言われるほどの「大化け」だった。

今期のクリエイターのほとんどが開発したシステムに非常にキャッチーな名前をつけた。図-22 にそれを示す。上のほうは、もうロゴデザインと言っていいレベルである。

(2016年6月26日受付)

竹内郁雄 (正会員) ■ nue@nue.org

1971年東京大学大学院修了。以降、NTT研究所、電気通信大学、東京大学、早稲田大学を経て現在、未踏IT人材発掘・育成事業統括プロジェクトマネージャ、一般社団法人未踏代表理事、東京大学名誉教授、本会フェロー。