Visionsketch: Gesture-based Language for End-user Computer Vision Programming

JUN KATO

IGARASHI LAB., THE UNIVERSITY OF TOKYO

http://junkato.jp/visionsketch/

東京大学
THE UNIVERSITY OF TOKYO

WHAT IS VISIONSKETCH?

Visionsketch language allows end-users
(= people without knowledge of programming)

- **to extract useful information from images/videos**
- **to make programs that can detect interesting events from live camera input**

by

- **building image processing pipelines**
- **with drawing shapes and choosing primitives**
- **without typing text**

WHAT IS VISIONSKETCH?

**Visionsketch language allows end-users
(= people without knowledge of programming)**

- **to extract useful information from images/videos**
- **to make programs that sting events from live ca**

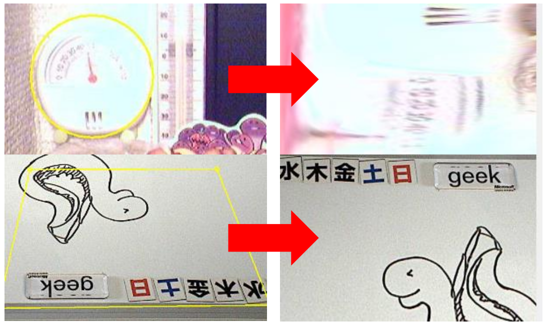
by

- **building in processing pipelines**
- **with drawing shapes and choosing primitives**
- **without typing text**

Let’s go visual!
a quick demo follows

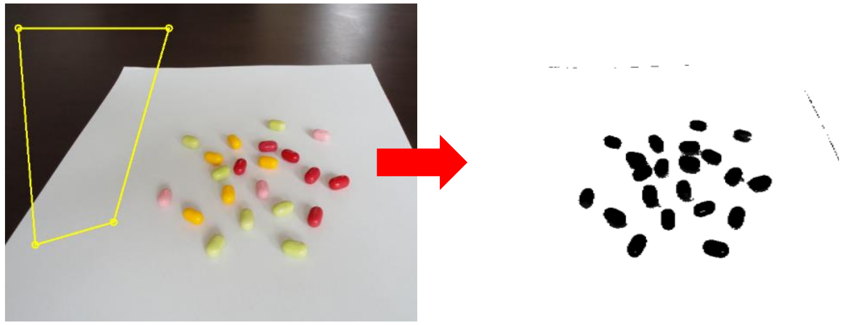
VISIONSKETCH LANGUAGE PRIMITIVES
DESIGNED ACCORDING TO USER INTERVIEWS

Geometric transform



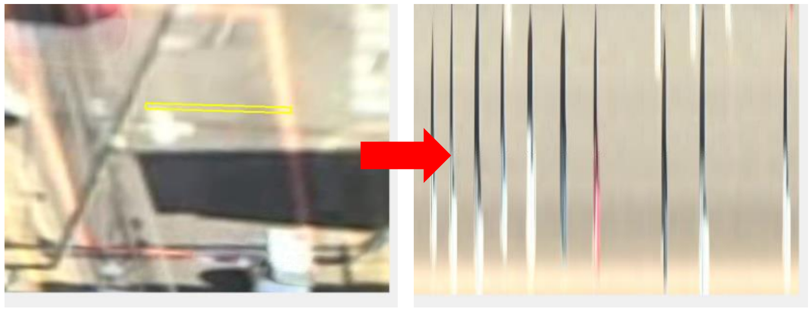
in: any image, **out**: image of same type

Information filtering



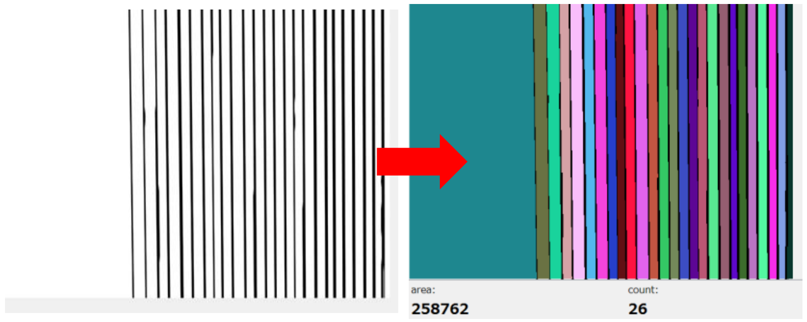
in: any image, **out**: bin image

Timelapse conversion



in: any image, **out**: image of same type

Contour counting



in: bin image, **out**: image + numbers

VISIONSKETCH LANGUAGE PRIMITIVES

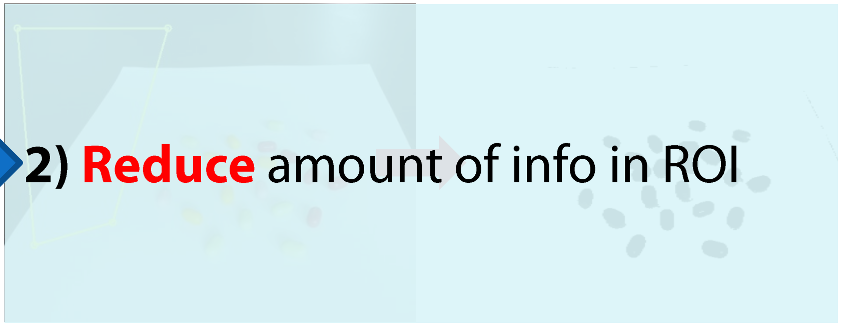
DESIGNED ACCORDING TO USER INTERVIEWS

Geometric transform

1) **Deform** region of interest (ROI) to make further processing easier

in: any image, **out**: image of same type

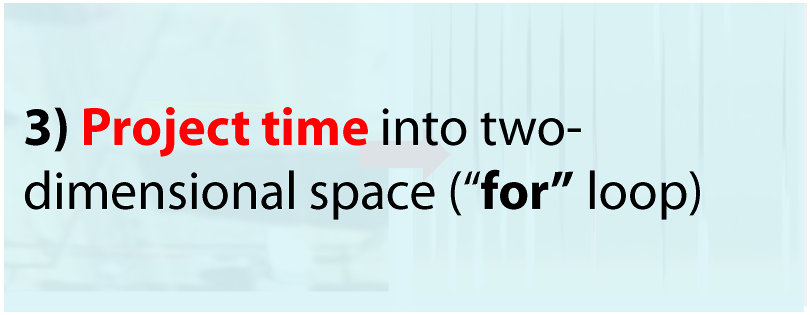
Information filtering



2) **Reduce** amount of info in ROI

in: any image, **out**: bin image

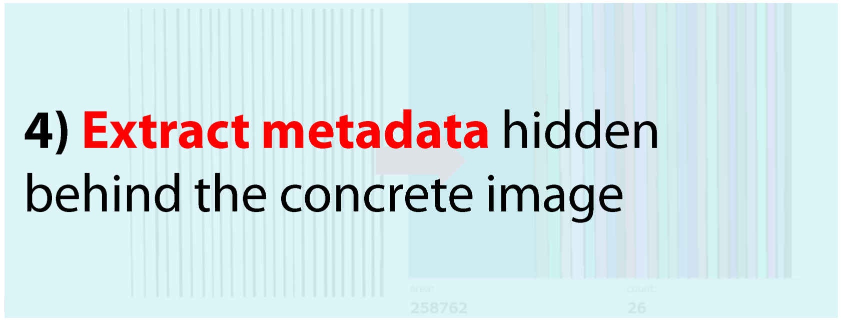
Timelapse conversion



3) **Project time** into two-dimensional space (“**for**” loop)

in: any image, **out**: image of same type

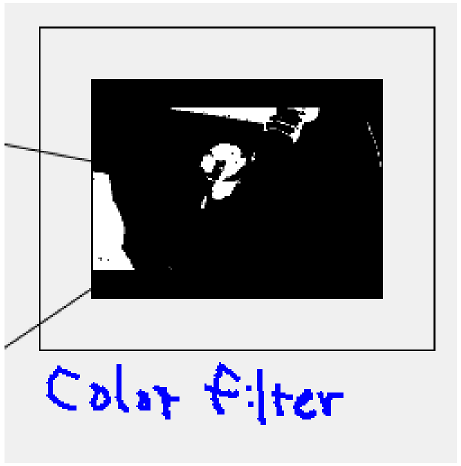
Contour counting



4) **Extract metadata** hidden behind the concrete image

in: bin image, **out**: image + numbers

SHARING SOME CONCEPTS WITH TEXT-BASED IDE

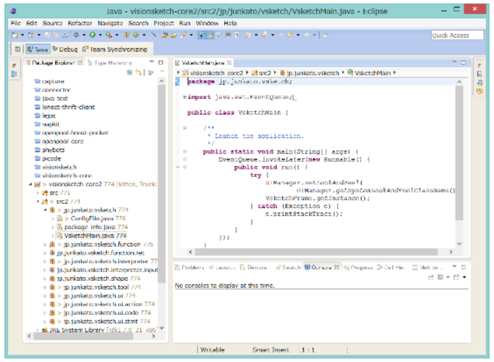


Comments in code

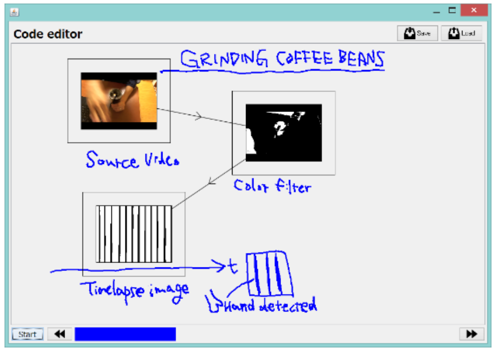
Text comment →

Freehand annotation

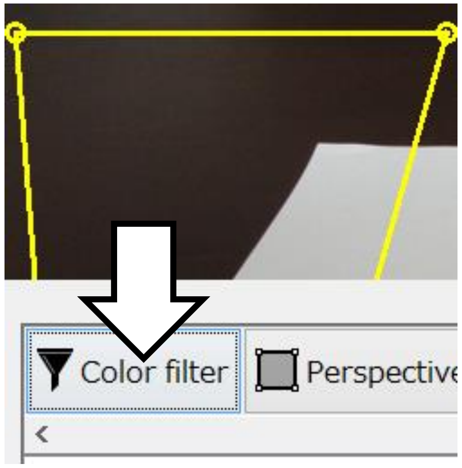
```
// Applying color filter
doSomeTasks();
```



Text-based IDE



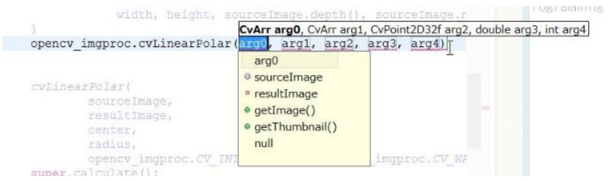
Visionsketch IDE



Code completion

Type-based completion →

Paramter-based completion



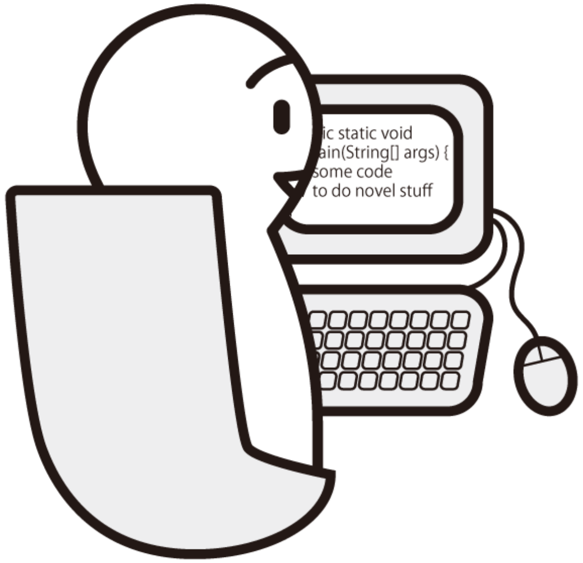
```
        width, height, sourceImage.depth(), sourceImage.r
    }
    opencv_imgproc.cvLinearPolar(arg0, arg1, arg2, arg3, arg4);

    cvLinearPolar(
            sourceImage,
            resultImage,
            center,
            radius,
            opencv_imgproc.CV_INT                    imgproc.CV_WA
    super.calculate();
```

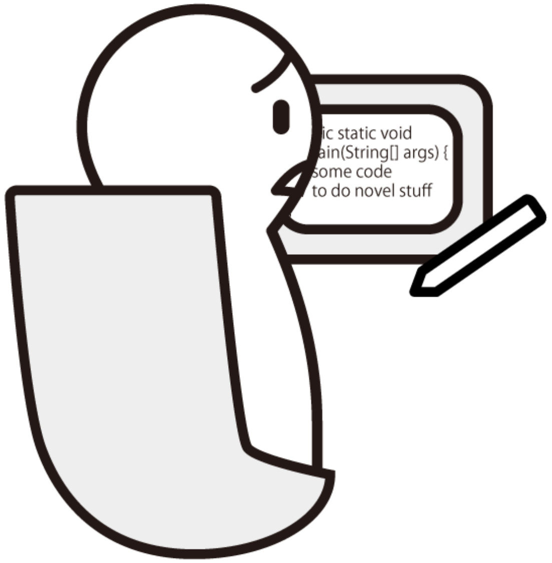
```
CvArr arg0, CvArr arg1, CvPoint2D32f arg2, double arg3, int arg4
arg0
sourceImage
resultImage
getImage()
getThumbnail()
null
```

PROGRAMMING LANGUAGE FOR "PEN & TOUCH" ERA?

Text is a good way to write program with a keyboard.



With pen & touch, we can't input text as before.

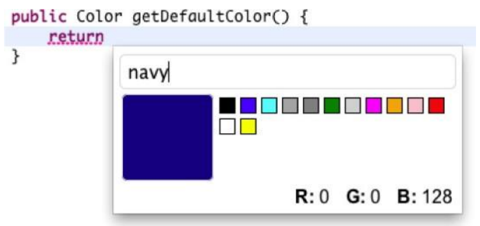


While **TouchDevelop** does good work with its software keyboard… ☺

RELATED WORK (1)
VISUAL REPRESENTATIONS IN IDE

Concrete data integrated in programming environment

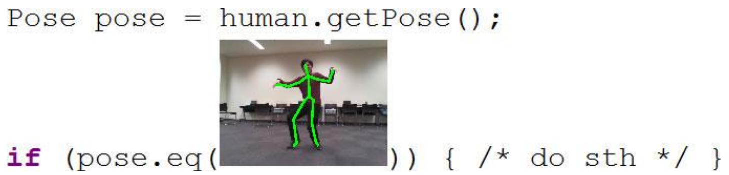
Active Code Completion
[Omar et al., ACM/IEEE ICSE '12]



```
public Color getDefaultColor() {
    return
}
```

Code completion enhancement

Picode IDE [Kato et al., ACM CHI'13]

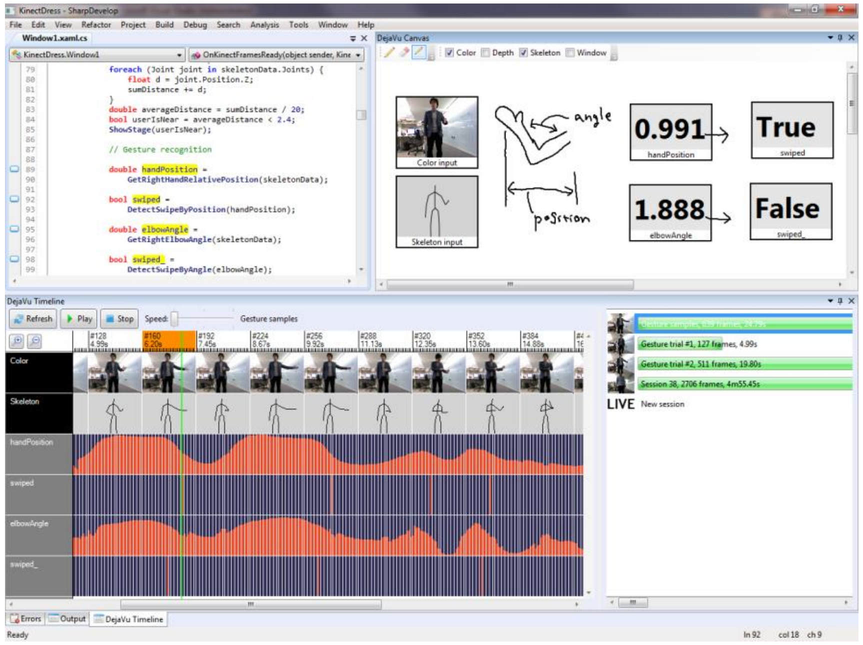


```
Pose pose = human.getPose();

if (pose.eq(      )) { /* do sth */ }
```

Code editor enhancement

DejaVu IDE [Kato et al., ACM UIST'12]



```
foreach (Joint joint in skeletonData.Joints) {
    float d = joint.Position.Z;
    sumDistance += d;
}
double averageDistance = sumDistance / 20;
bool userIsNear = averageDistance < 2.4;
ShowStage(userIsNear);

// Gesture recognition

double handPosition =
    GetRightHandRelativePosition(skeletonData);

bool swiped =
    DetectSwipeByPosition(handPosition);

double elbowAngle =
    GetRightElbowAngle(skeletonData);

bool swiped_ =
    DetectSwipeByAngle(elbowAngle);
```

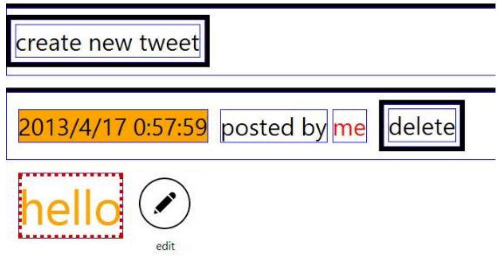
KinectDress - SharpDevelop
File Edit View Refactor Project Build Debug Search Analysis Tools Window Help
Window1.xaml.cs
KinectDress.Window1
OnKinectFramesReady(object sender, Kine
DejaVu Canvas
Color Depth Skeleton Window
Color input
Skeleton input
angle
position
0.991
handPosition
True
swiped
1.888
elbowAngle
False
swiped_
DejaVu Timeline
Refresh Play Stop Speed: Gesture samples
Color
Skeleton
handPosition
swiped
elbowAngle
swiped_
Gesture trial #1, 127 frames, 4.99s
Gesture trial #2, 511 frames, 19.80s
Session 38, 2706 frames, 4m55.45s
LIVE New session
Errors Output DejaVu Timeline
Ready

Debugger enhancement

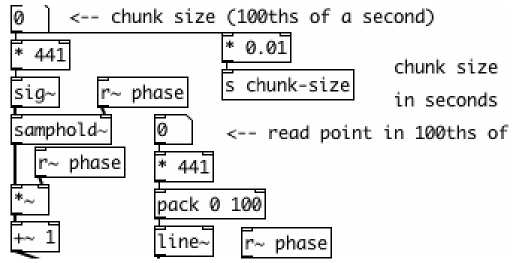
RELATED WORK (2)
LIVE PROGRAMMING

Direct manipulation of program

TouchDevelop for **GUI**

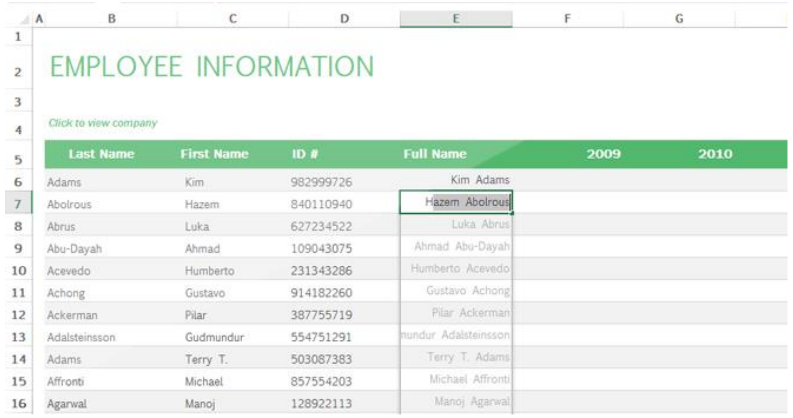


PureData for **audio processing**



```
0   <-- chunk size (100ths of a second)
* 441                * 0.01
                                 chunk size
sig~     r~ phase    s chunk-size
                                 in seconds
samphold~        0    <-- read point in 100ths of
  r~ phase       * 441
*~               pack 0 100
+~ 1             line~    r~ phase
```

Excel for **spreadsheet calculation**



EMPLOYEE INFORMATION

Click to view company

Last Name	First Name	ID #	Full Name	2009	2010
Adams	Kim	982999726	Kim Adams		
Abolrous	Hazem	840110940	Hazem Abolrous		
Abrus	Luka	627234522	Luka Abrus		
Abu-Dayah	Ahmad	109043075	Ahmad Abu-Dayah		
Acevedo	Humberto	231343286	Humberto Acevedo		
Achong	Gustavo	914182260	Gustavo Achong		
Ackerman	Pilar	387755719	Pilar Ackerman		
Adalsteinsson	Gudmundur	554751291	nundur Adalsteinsson		
Adams	Terry T.	503087383	Terry T. Adams		
Affronti	Michael	857554203	Michael Affronti		
Agarwal	Manoj	128922113	Manoj Agarwal		

Word? for **HTML+CSS editing?**

FROM USER INTERFACE TO PROGRAMMING LANGUAGE

There is smooth gradation rather than deep valley.

User interface = programming language? — **No**. There's no abstraction.

HTML = programming language? — Probably… **no**? It's not turing-complete. (While HTML + CSS3 are! ☺)

Visionsketch = programming language?

SQL = programming language? — **Yes**, while the domain is limited.

C, C++, Java, … = programming language? — Definitely **yes**!

MELTING THE BOUNDARY BETWEEN UI AND PL

My research contributions:

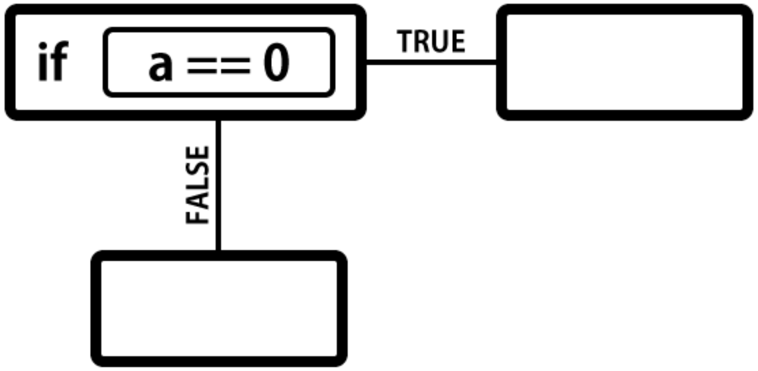
- **Live programming of image processing programs with visual representations**
- **Bringing UI perspective to PL (User-centered design of touch-optimized language)**
- **Exporting PL techniques to UI world (Language primitives, IDE, code completion…)**

User interface and programming language are both computational languages and can share many stuff ☺

APPENDIX

RELATED WORK (0)
VISUAL PROGRAMMING LANGUGE

Symbolic representations of program code



These do not fully benefit from pen & touch… code elements are still something symbolic.

We should be able to draw something concrete.

FROM USER INTERFACE TO PROGRAMMING LANGUAGE

Every one of these is language = {syntax + words}

- **to make the computer work for us**
- **designed to balance easiness and degree-of-freedom**

User interface = language	GUI components + possible operations
HTML = language	HTML spec + HTML tags
SQL = language	Syntax + statements
C, C++, Java, … = language	Syntax + statements