

Ferramenta para transcrição do alfabeto datilológico para texto utilizando Microsoft Kinect

Diego Marcelo Santin

Orientador: Aurélio Faustino Hoppe

Motivação

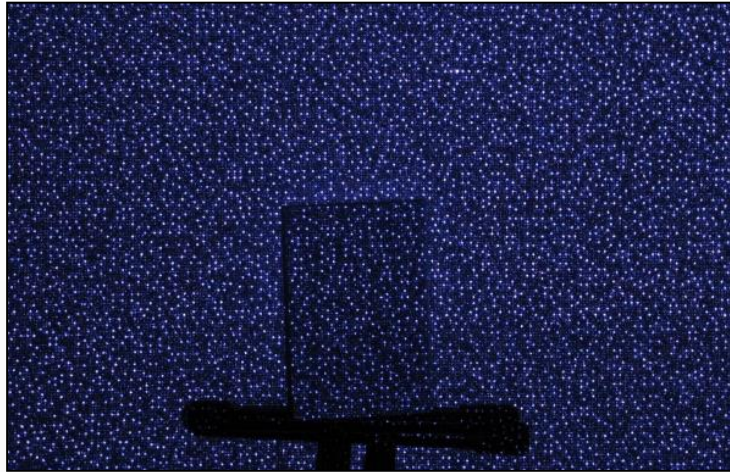
- Acessibilidade dos deficientes auditivos
- Adaptação de novas tecnologias para pessoas com necessidades especiais
- Utilização do sensor Microsoft Kinect para captura de movimentos

Fundamentação

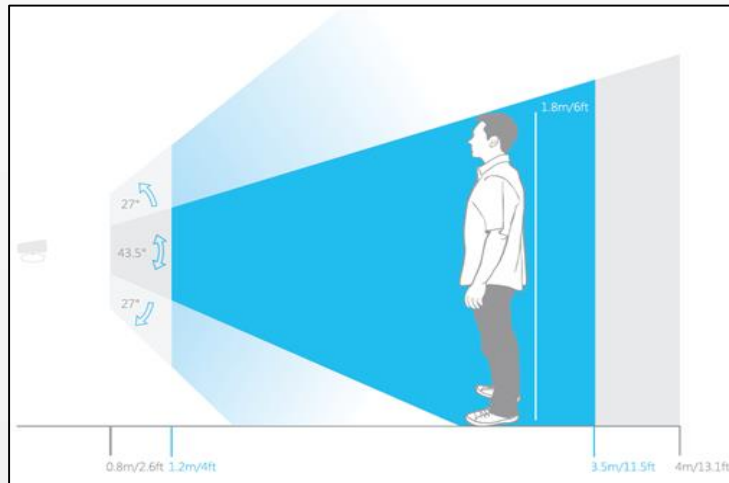
- LIBRAS



Fundamentação

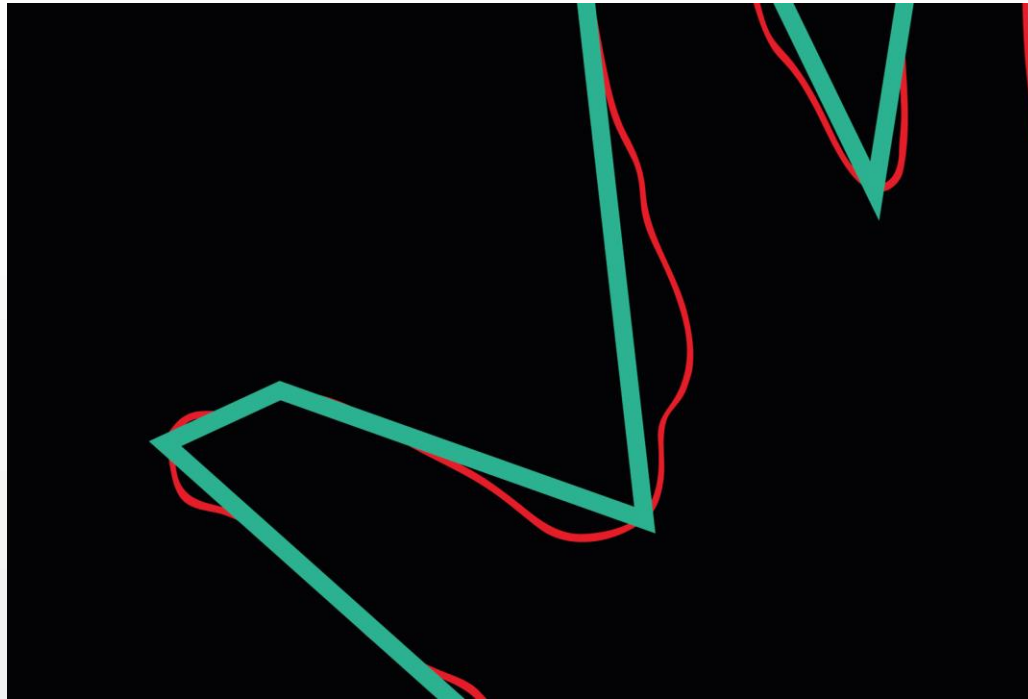


- Captura de dados de profundidade
- Uso da câmera RGB
- Motor de inclinação
- Influência da luz

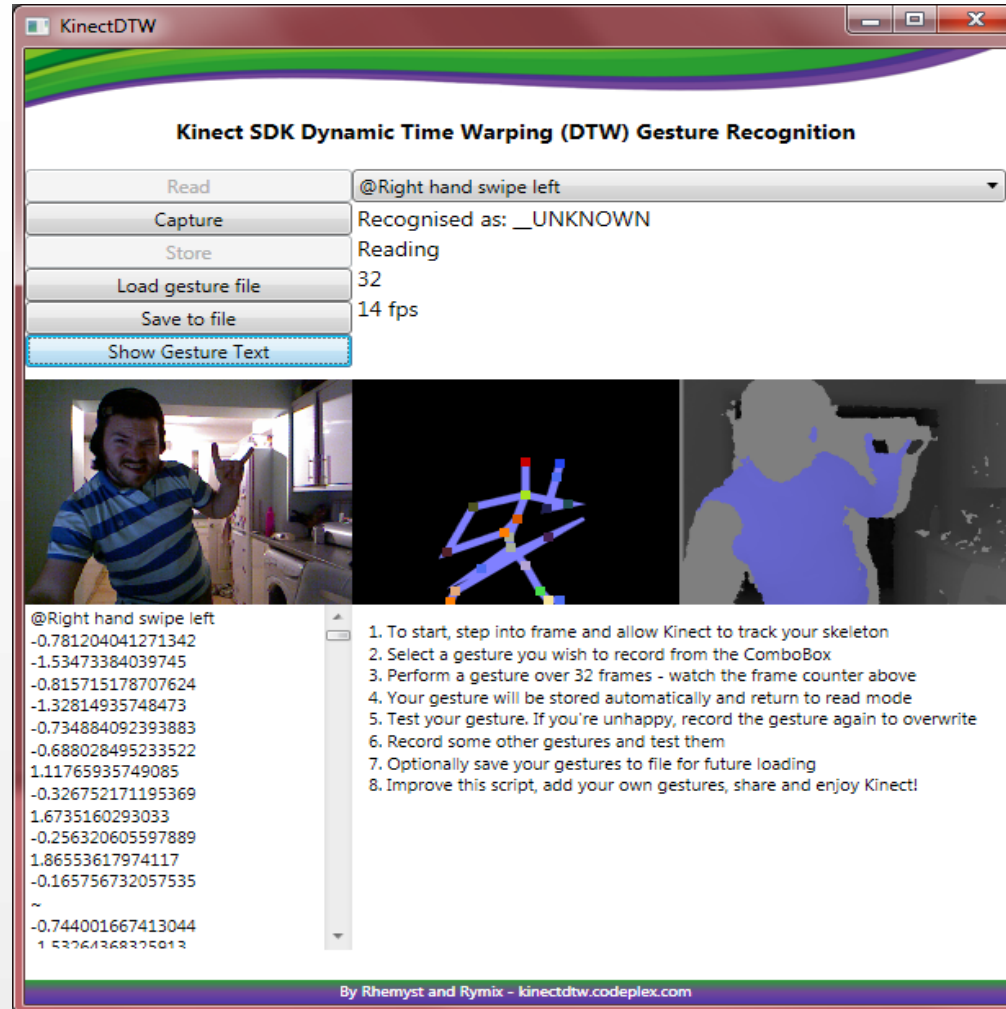


Fundamentação

- Algoritmo Ramer-Douglas-Peucker


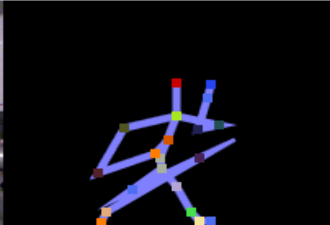



Trabalhos relacionados – Kinect DTW



Kinect SDK Dynamic Time Warping (DTW) Gesture Recognition

Read: @Right hand swipe left
Capture: Recognised as: _UNKNOWN
Store: Reading
Load gesture file: 32
Save to file: 14 fps
Show Gesture Text

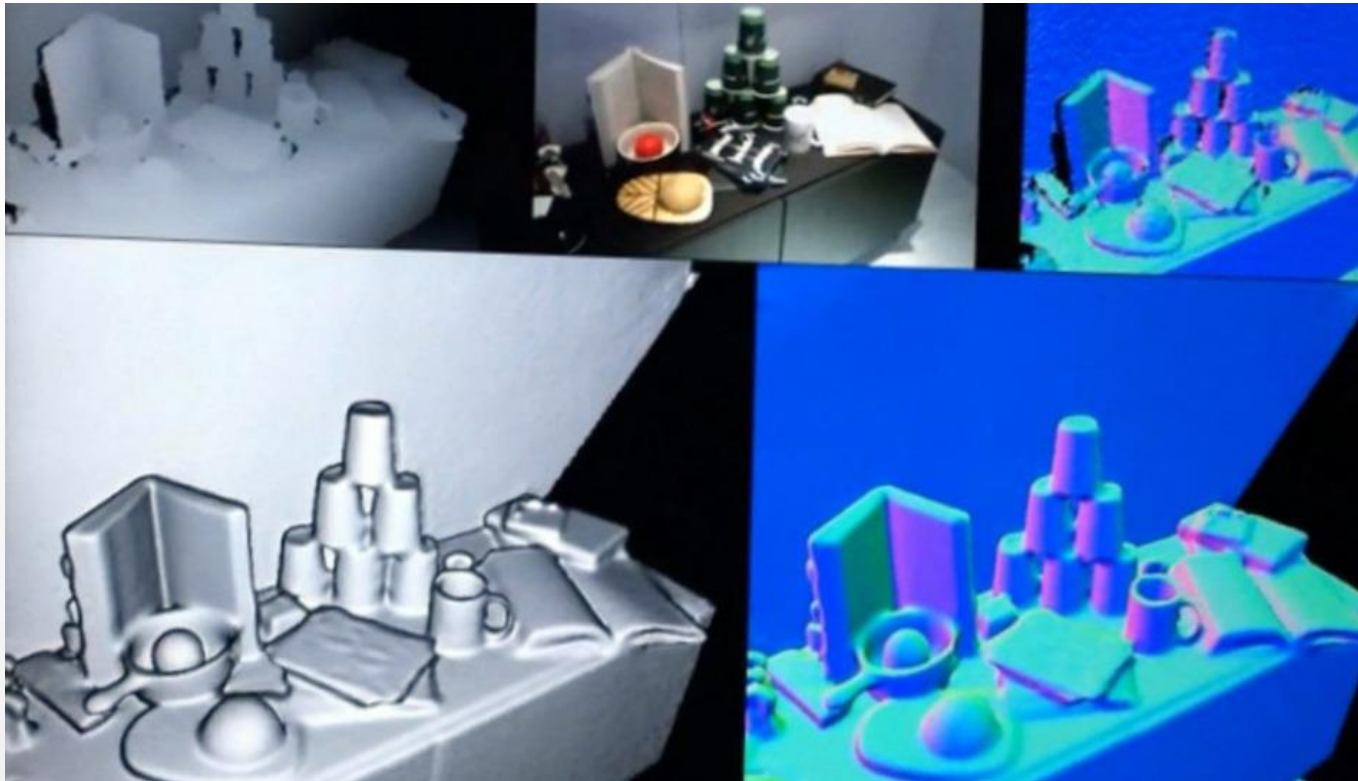
  

@Right hand swipe left
-0.781204041271342
-1.53473384039745
-0.815715178707624
-1.32814935748473
-0.734884092393883
-0.688028495233522
1.11765935749085
-0.326752171195369
1.6735160293033
-0.256320605597889
1.86553617974117
-0.165756732057535
~
-0.744001667413044
1 5 3 2 6 4 1 3 6 8 3 7 5 0 1 2

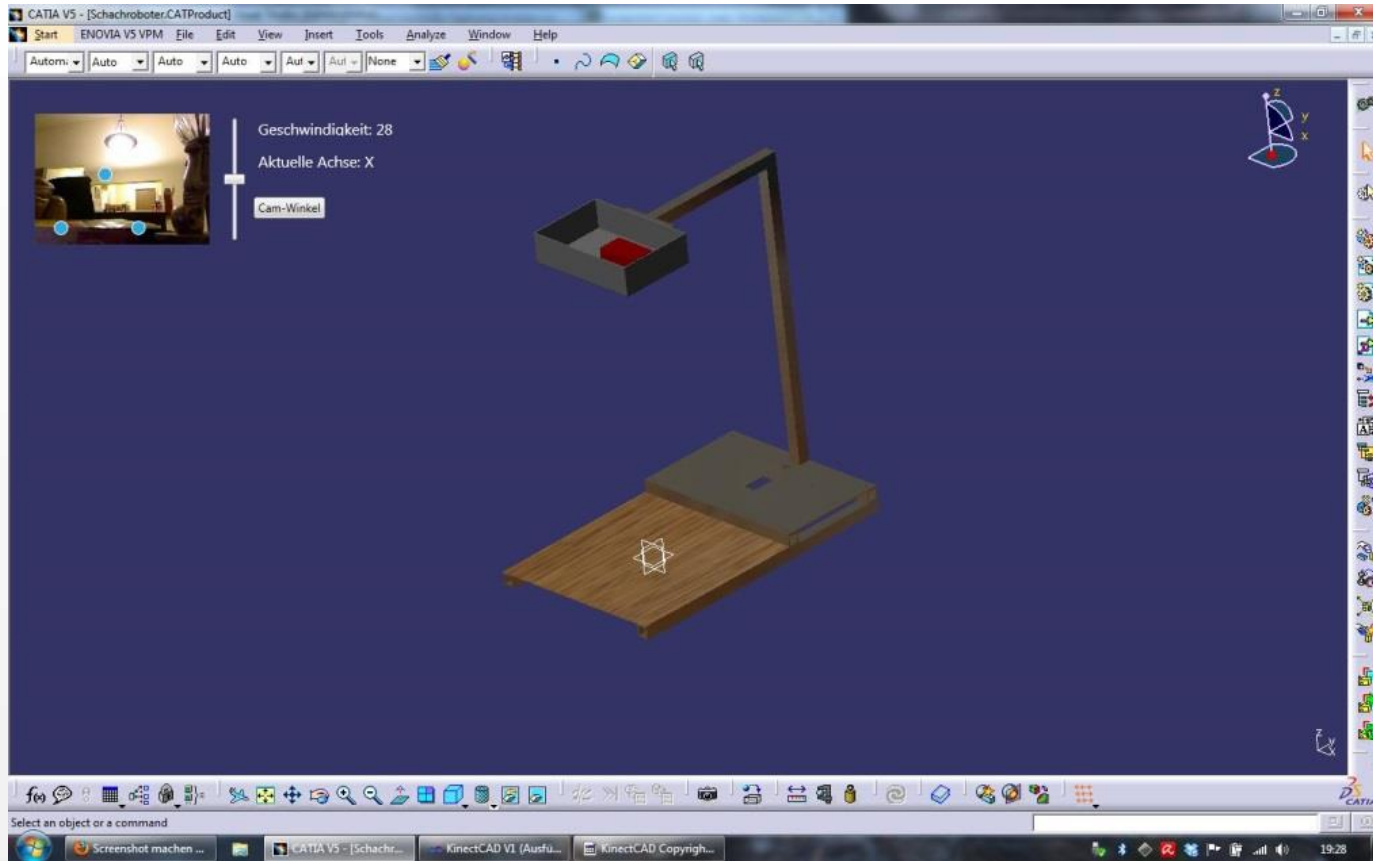
1. To start, step into frame and allow Kinect to track your skeleton
2. Select a gesture you wish to record from the ComboBox
3. Perform a gesture over 32 frames - watch the frame counter above
4. Your gesture will be stored automatically and return to read mode
5. Test your gesture. If you're unhappy, record the gesture again to overwrite
6. Record some other gestures and test them
7. Optionally save your gestures to file for future loading
8. Improve this script, add your own gestures, share and enjoy Kinect!

By Rhemyst and Rymix - kinectdtw.codeplex.com

Trabalhos relacionados – Kinect Fusion



Trabalhos relacionados - KinectCAD



Trabalhos relacionados

característica / trabalho relacionado	Kinect DTW	Kinect Fusion	KinectCAD
utiliza <i>skeletal tracking</i>	X		X
utiliza <i>stream</i> de vídeo	X	X	X
utiliza <i>stream</i> de profundidade		X	
utiliza Kinect <i>for Windows SDK</i>	X	X	X
faz reconhecimento de gestos	X		X
principal funcionalidade utilizada da Kinect <i>for Windows SDK</i>	<i>skeletal tracking</i>	<i>stream</i> de profundidade	<i>stream</i> de profundidade

Trabalho proposto

- Capturar os gestos do usuário utilizando os sensores do Kinect
- Reconhecer os gestos baseando-se em um banco pré-definido com o alfabeto datilológico
- Transcrever o movimento realizado para a correspondência em texto

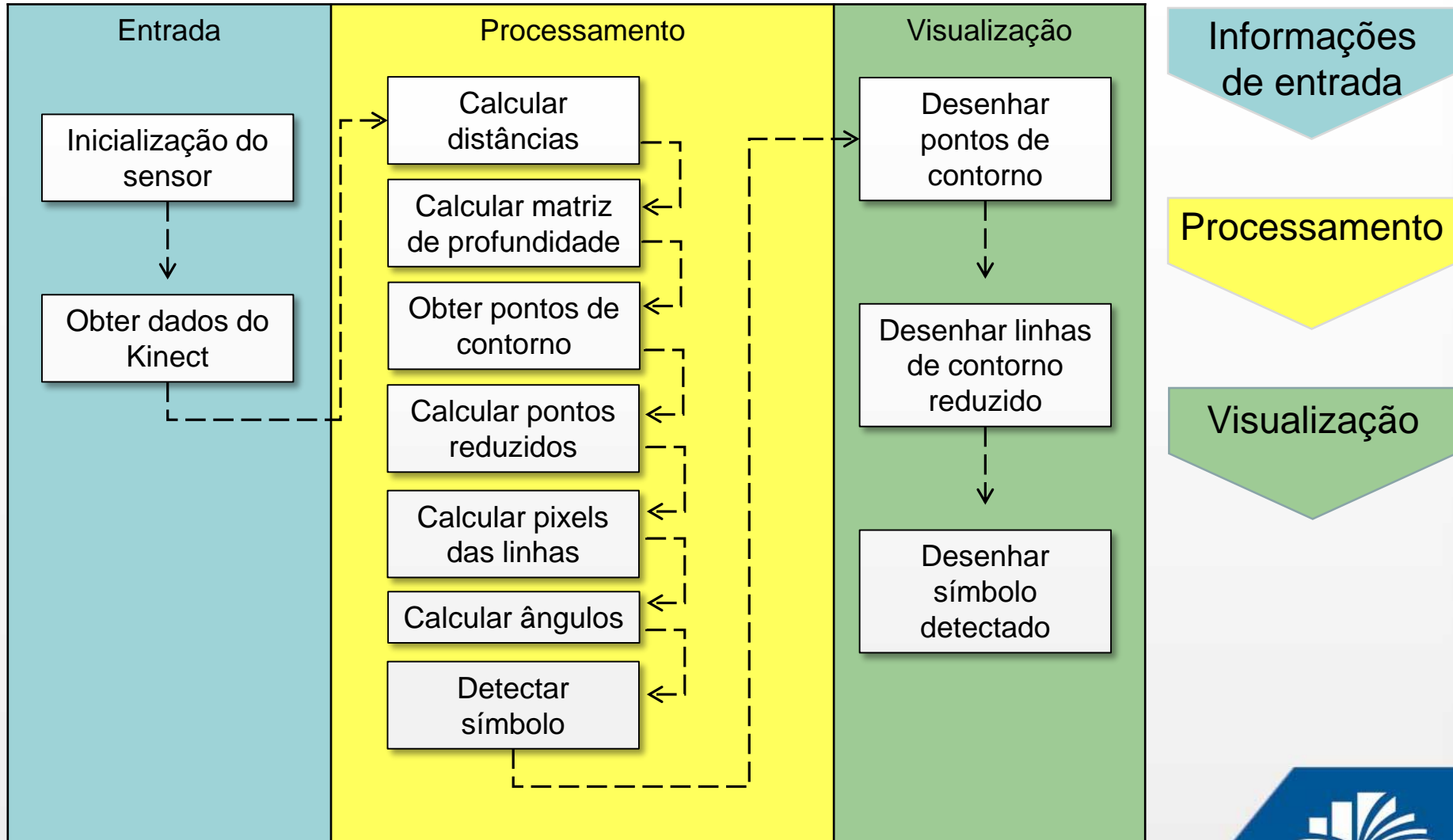
Requisitos

- Permitir ao usuário informar qualquer letra e/ou número contidos no alfabeto datilológico (RF)
- Transcrever a entrada do usuário para texto na tela do computador (RF)
- Utilizar um arquivo XML contendo dados sobre as formas geométricas para comparação com os sinais de entrada (RF)

Requisitos

- Ser implementado utilizando a linguagem de programação C# (RNF)
- Ser compatível com sistemas operacionais que ofereçam suporte ao .NET Framework (RNF)
- Utilizar o sensor de movimentos Microsoft Kinect (RNF)
- Utilizar o Kinect *for* Windows SDK (RNF)

Etapas da transcrição

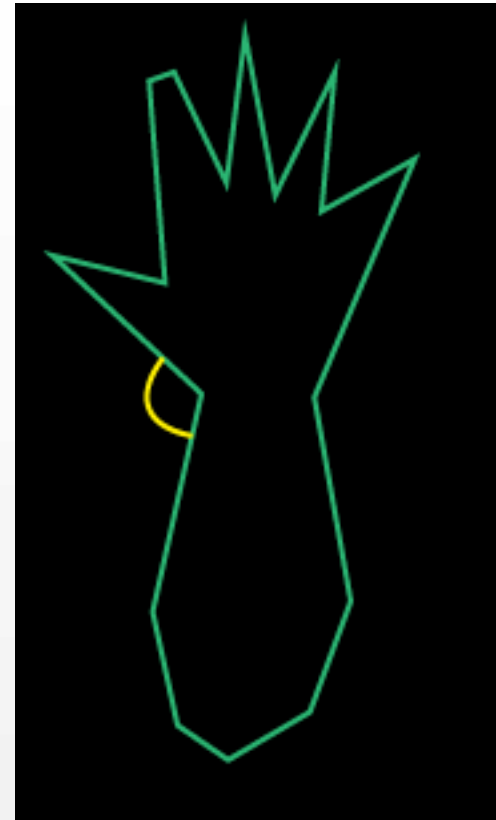


Desenvolvimento: visão final



Informação de entrada

```
<SymbolData
  <Symbols>
    <Symbol character="53">
      <SymbolPoints>
        <SymbolPoint angle="40" />
        <SymbolPoint angle="15" />
        <SymbolPoint angle="15" />
        <SymbolPoint angle="20" />
        <SymbolPoint angle="30" />
        <SymbolPoint angle="35" />
      </SymbolPoints>
    </Symbol>
  </Symbols>
</SymbolData>
```

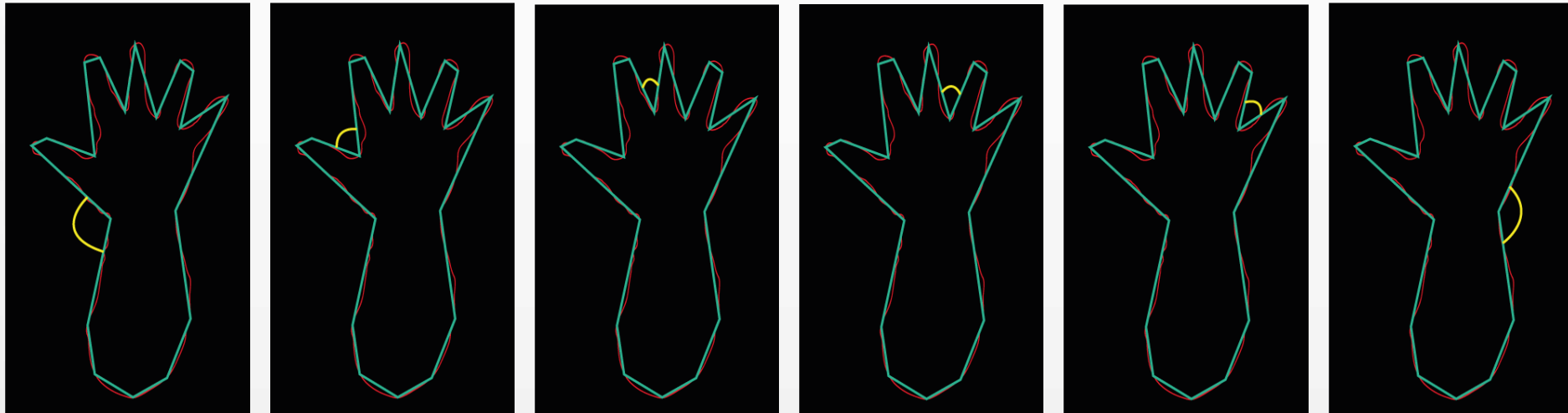


Comparação dos ângulos

```
angle >= (point.Angle-GAP) && angle <= (point.Angle+GAP)
```

```
angleComparer comparer(currentPoint);
```

```
angulo = std::find_if(angles.begin(), angles.end(), comparer);
```



Ângulos processados

Ângulo em radianos:

```
Vector2D<double>::angle(p2-p1, p3-p1);
```

Produto interno:

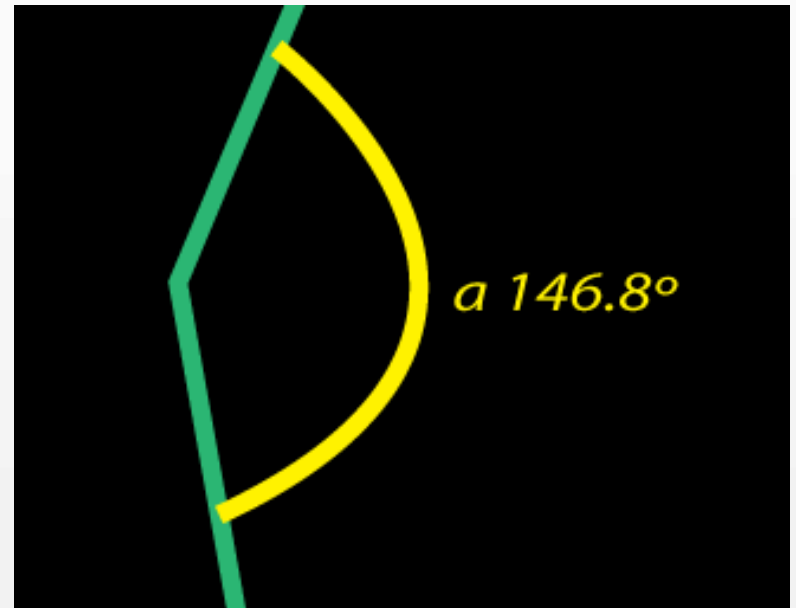
```
math::acos( (v1 * v2) / (v1.norm() * v2.norm()) );
```

Normal:

```
math::sqrt(x*x + y*y);
```

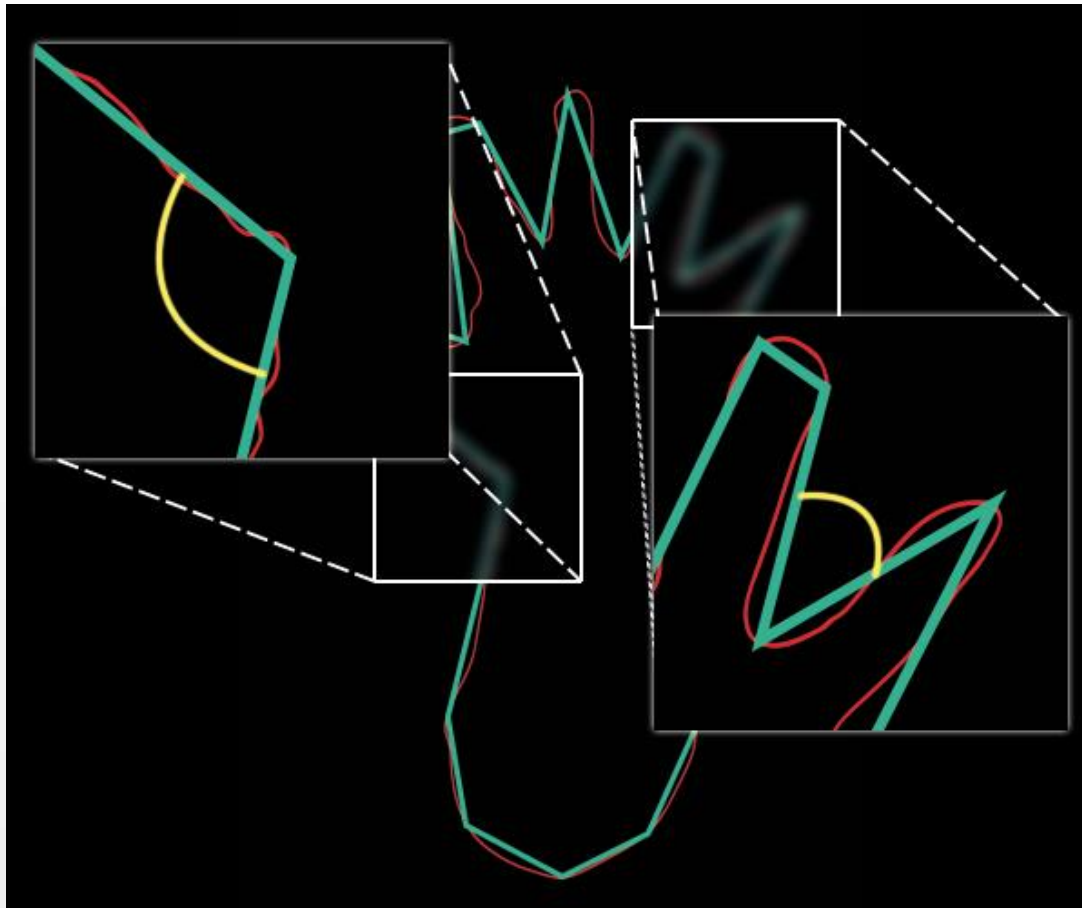
Valor em graus:

```
rad * 180.0 / M_PI;
```

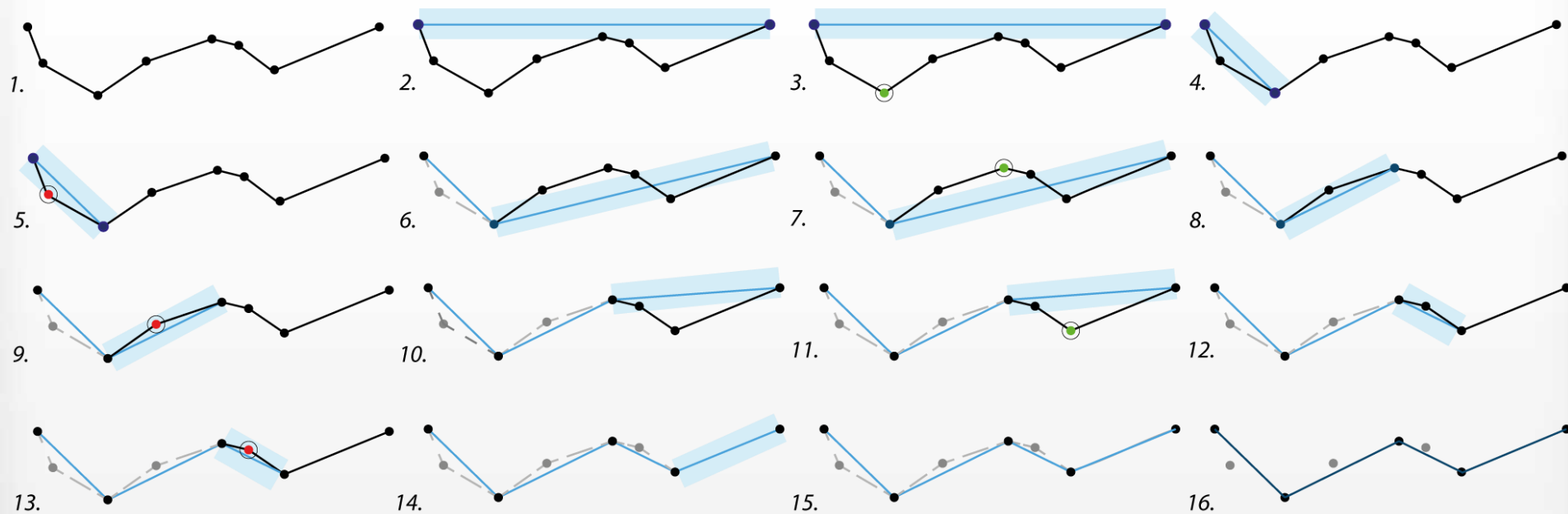


Detecção dos ângulos

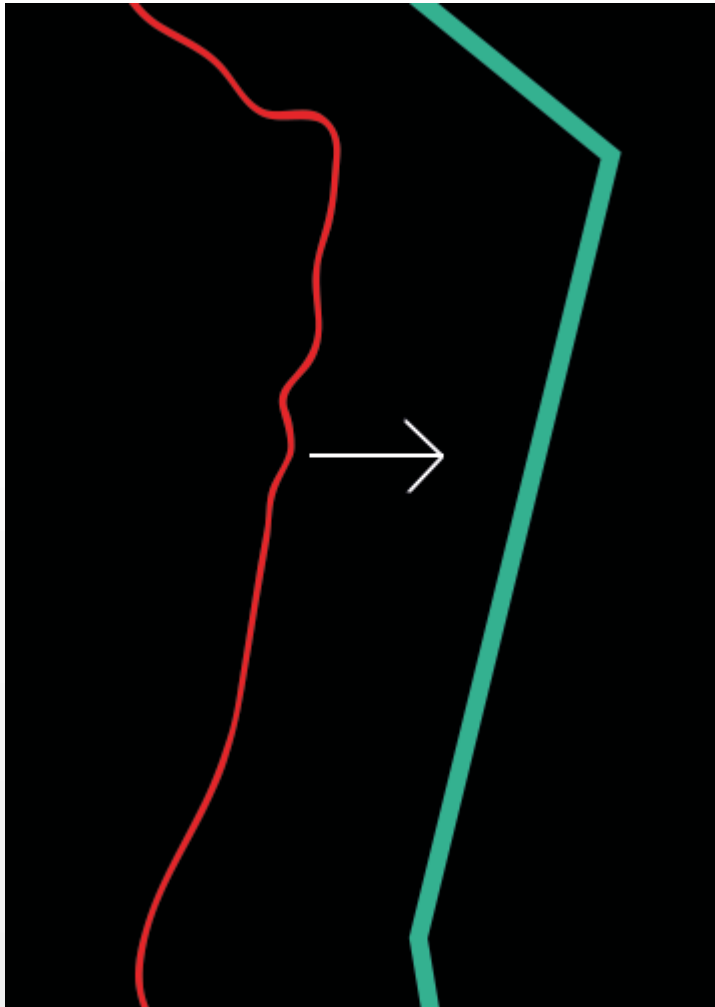
- Executada em todos pontos do contorno
- Para cada ponto existente, um ângulo é formado é pelas linhas adjacentes ao ponto



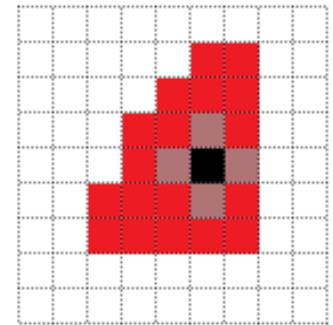
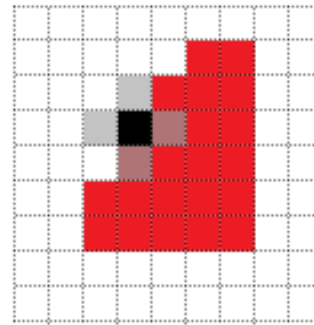
Aplicação de Ramer-Douglas-Peucker



Redução das linhas de contorno



- Busca de pontos adjacentes
- Cálculo da fronteira



Distâncias

- Uso do sensor de profundidade
- Cálculo efetuado utilizando *bitshift*



Captura dos dados

- Leitura dos dados do sensor de profundidade, utilizados para criação do contorno do usuário
- Leitura dos dados da câmera RGB, utilizados para representar o *feedback* dado ao usuário



Testes

- Testes efetuados utilizando 27 dos 35 símbolos presentes no alfabeto datilológico
- Símbolos cadastrados separadamente e em paralelo
- Símbolos cadastrados com ângulos côncavos e convexos, ou apenas ângulos côncavos
- Percentual de acerto computado por seção
- Alteração de C# para C++

RESULTADOS DE TRANSCRIÇÃO

Símbolos cadastrados separadamente –
70%

Símbolos cadastrados em paralelo –
11%

0	OK	G	OK	0	NÃO OK	G	NÃO OK
1	OK	I	OK	1	NÃO OK	I	NÃO OK
2	OK	L	OK	2	NÃO OK	L	NÃO OK
3	OK	M	S-OK	3	NÃO OK	M	NÃO OK
4	OK	N	S-OK	4	NÃO OK	N	NÃO OK
5	S-OK	P	OK	5	NÃO OK	P	NÃO OK
6	OK	Q	S-OK	6	NÃO OK	Q	NÃO OK
7	OK	R	OK	7	NÃO OK	R	NÃO OK
9	S-OK	T	OK	9	NÃO OK	T	NÃO OK
A	OK	U	S-OK	A	NÃO OK	U	S-OK
B	S-OK	V	OK	B	NÃO OK	V	S-OK
C	S-OK	Y	OK	C	NÃO OK	Y	S-OK
D	OK			D	NÃO OK		
E	OK			E	NÃO OK		
F	OK			F	NÃO OK		

OK = Identificado com sucesso. **NÃO OK** = Não identificado. **S-OK** = Com dificuldades

Conclusões

- Processamento satisfatório para detecção de contornos
- Resultado satisfatório para detecção de símbolos, cadastrados separadamente
- DLL de processamento pode ser utilizada em outras plataformas

Limitações

- Operado apenas em plataformas que suportam .NET.
- Detecção exclusiva de símbolos estáticos
- Redução no reconhecimento dos símbolos quando cadastrados simultaneamente
- Distância limitada para total funcionalidade do software
- Forma única de execução do símbolo para reconhecimento

Extensões

- Aumentar a quantidade de ângulos descritos em cada símbolo e aprimorar a forma comparação utilizada
- Executar testes com um público alvo maior para estudar a viabilidade de implantação do software como um serviço público e gratuito
- Emitir um som correspondente ao símbolo detectado para aprimorar o treinamento do usuário
- Armazenar todos caracteres transcritos com o objetivo de possibilitar a escrita de uma frase
- Implementar detecção de símbolos que exijam movimentos

Demonstração

Obrigado!