09

Proposta de técnica para segmentação com base em histograma para imagens de diferentes resoluções adquiridas com Kinect

Alex Luis da Costa Alexandre
Universidade Estadual do Maranhão
alexaer973@gmail.com|LATTES

Mauro Sérgio Silva Pinto
Universidade Estadual do Maranhão
maurosergiospinto@gmail.com| ORCID

Denner Robert Rodrigues Guillon
Universidade Estadual do Maranhão
dennerguillhon@gmail.com

Recebido em: 17/03/2022
Aprovado em: 12/05/2025

DOI: http://dx.doi.org/10.5965/198431782112025e0073
eLocation-id: e0073

Proposta de técnica para segmentação com base em histograma para imagens de diferentes resoluções adquiridas com Kinect

O trabalho apresenta uma forma alternativa na comunicação entre pessoas que só conseguem se comunicar pela linguagem brasileiras de sinais (LIBRAS) e pessoas que não tem domínio da língua. A maneira escolhida para facilitar esta comunicação é com a utilização do sensor Kinect (usado em vídeo game Xbox 360 e One) que dispõem de tecnologias de captura de imagens em RGB e imagens em profundidade. Estas facilitam o processamento em software como Matlab. No trabalho são apresentadas técnicas de aquisição, segregação e extração de informações das imagens além de sua aplicação em conjunto de atores com idades, sexo, tonalidade da pele e estaturas diferentes. Em diversos trabalhos são apresentadas técnicas de segmentação usando as imagens de profundidade. No sensor Kinect 360 isso é relativamente simples de realizar, já o sensor Kinect One apresenta dificuldades na aplicação, devido a diferença nas resoluções. Nosso trabalho trás uma abordagem que resolve de forma satisfatória este problema. Este trabalho também apresenta uma análise bibliográfica das publicações mais utilizadas. Ele auxilia pesquisadores iniciantes com as técnicas e os comandos utilizados no Matlab.

Palavras-chave:: LIBRAS; Kinect; Segmentação.

Proposed histogram-based segmentation technique for different resolution images acquired with Kinect

This work presents an alternative form of communication between people who can only communicate through the Brazilian language of signs and people who do not have command of the language. The way presented is with the use of the sensor Kinect (used in video game Xbox one) which have image capture technologies that facilitate processing in software such as Matlab. This work presents techniques for the acquisition, segregation and extraction of information from the images, as well as their application to actors of different ages, gender, skin tone and height. In several works are presented segmentation techniques using profundidade images. In Kinect 360 sensor this is relatively simple to accomplish, while Kinect One sensor has difficulties in application due to the difference in resolutions. Our work takes an approach that satisfactorily solves this problem. This work also presents a bibliographical analysis of the most used publications. It assists novice researchers with the techniques and commands used in Matlab.

Keywords: LIBRAS; Kinect; Segmentation.

INTRODUÇÃO

Na comunicação gestual brasileira temos um grupo de pessoas com deficiência auditiva que possuem uma língua própria, chamada Língua Brasileira de Sinais ou simplesmente LIBRAS. A língua de sinais não é igual ao português, tem morfologia e sintaxe próprias, é um idioma brasileiro independente. Como qualquer outra língua, é preciso estudar a gramática e estruturação da frase para dominar esta língua. A LIBRAS é reconhecida por lei como forma de comunicação e expressão das comunidades surdas do Brasil.

Os avanços das tecnologias com relação a captura de imagens têm viabilizado técnicas de visão computacional e reconhecimento de padrões em diferentes áreas da atividade humana (MENDONÇA, 2013). Gestos e expressões corporais emitidas pelo homem ainda não são igualmente assimiláveis pelos atuais sistemas de computação. Nesse sentido faz-se necessário trabalhos de pesquisas de hardware e software para áreas como a interação humano-computador ou de aplicação de interfaces naturais com o usuário (JUNIOR, 2014). Um sistema artificial de visão é um sistema computadorizado capaz de adquirir, processar e interpretar imagens em tempo real (FILHO; NETO, 1999). A Figura 1 mostra um diagrama de blocos de uma SVA.

Figura 1: Um Sistema de Visão Artificial (SVA) e suas principais etapas

Fonte: (FILHO; NETO, 1999)

1.1. ETAPAS

1.1.1. Aquisição

Uma das etapas mais importantes é a captura de imagem, nesta etapa consiste na entrada de dados que serão tratados e analisados para o processamento da imagem.

O rastreio das mãos não é uma tarefa fácil, as mãos são objetos que se deformam mudando de posição no espaço. O movimento das mãos pode ocultar o movimento da outra e da face. Imagens de profundidade são utilizadas para diminuir este problema (CORREIA, 2013).

A proposta é criar uma interação humano-computador (IHC) utilizando recursos de visão computacional, sem utilização de dispositivos como teclado, mouse.

Para aquisição de imagens será utilizado o sensor Microsoft Kinect utilizado no videogame Xbox. O kinect tem dois canais de vídeo: imagens RGB e imagens de profundidade. Para realizar o processamento de imagem de profundidade, é utilizado o sensor infravermelho projetando uma imagem feita por padrões de difração pseudoaleatórios estáticos (um holograma gerado computacionalmente) sobre a cena. (JUNIOR, 2014).

1.1.2. Pré-processamento

Uma imagem pode apresentar diversas imperfeições, tais como: pixel ruidosos, contraste e/ou brilho inadequado, caracteres interrompidos. A função desta etapa é aprimorar a imagem para as fases subsequentes. As operações efetuadas são de baixos níveis porque trabalham com a intensidade dos pixels. A imagem resultante é uma imagem de melhor qualidade que a original (FILHO; NETO, 1999).

1.1.3. Segmentação

Consiste em subdividir a imagem em regiões para facilitar a interpretação dela. A subdivisão é feita através do agrupamento de pixels utilizando características (features) comuns. Estas propriedades podem ser cores, intensidade, textura ou continuidade (ANJO, 2013).

A aplicação de filtros é umas das principais técnicas para extração de características de uma imagem sendo de grande importância segmentar uma região de interesse, o que reduz consideravelmente o tempo de processamento (PAVAN; CAZHURRIRO; MODESTO, 2010).

1.1.4. Extração de Características

Nesta etapa o objetivo é extrair características das imagens resultantes da segmentação através de descritores que possam representar cada digito e diferenciar os dígitos parecidos. Estes descritores devem ser representados por uma estrutura de dados adequada ao algoritmo de reconhecimento (FILHO; NETO, 1999).

1.1.5. Reconhecimento e Interpretação

A classificação é processo de extração de informações em objetos para reconhecer padrões e objetos. Associa cada pixel da imagem a um “rótulo” descrevendo um objeto real. Na classificação são extraídas da imagem informações mais convenientes à interpretação automática.

Em geral, refere-se ao agrupamento de dados em conjuntos similares. Esta informação é diversas vezes usada em passos de análises para qualquer sistema de processamento de sinal/dados. A classificação de imagem é similar a classificação de dados em geral, mas pode variar dependendo da aplicação em que é utilizada (QIDWAI; CHEN, 2009).

O reconhecimento é o processo de atribuição de um rótulo a um objeto tendo como base suas características. A interpretação consiste em atribuir um significado a um conjunto de objetos (FILHO; NETO, 1999).

1.2. TRABALHOS RELACIONADOS

Existem várias propostas para resolver o problema de reconhecimento de gestos. Podemos dividir estes trabalhos em dois grupos: um que usa sensores acoplados ao corpo e outro que usa visão computacional. Os que envolvem sensores acoplados ao corpo (exemplo as luvas) tendem a facilitar o processo de digitalização e resultados mais confiáveis, mas gerando desconforto ao usuário. Já os de visão computacional só necessitam de uma câmera, sendo a sua desvantagem a variação de iluminação (MONTEIRO et al, 2016).

Para reconhecimento de gestos (PAVAN; CAZHURRIRO; MODESTO, 2010) utiliza uma webcam para capturar imagens dinâmicas e extrair as características da mão segmentada. Tendo como objetivo a criação de letras ou sinais suficientes para criação de palavras simples.

(SOARES; RAIA, 2014) apresenta um dispositivo na forma de vestuário equipado com um sensor de profundidade Kinect da Microsoft para auxiliar pessoas com limitações visuais. No projeto foi colocado um indivíduo num espaço fechado com vários obstáculos, durante o deslocamento o indivíduo recebia informações constantes, trazidas por meio de transdutores vibratórios embutidos na vestimenta. O projeto teve problemas na identificação de objetos com superfícies altamente reflexiva ou polida.

Em diversas áreas tem sido aplicado redes neurais. (ALVARENGA; CORREA; OSÓRIO, 2012) propõem um sistema para o aprendizado e classificação de gestos com a utilização do sensor de profundidade Kinect sendo executado em tempo real. Sendo desenvolvido um programa supervisor em Python. O programa analisa a posição da mão direita, armazena pontos e deslocamentos em listas separadas. O programa foi desenvolvido para reconhecer três gestos: Circle; Come here e good bye.

Uma abordagem para classificação de um conjunto de nove sinais é proposta por (MENDONÇA, 2013) (entregar, pegar, abrir, olhar, empurrar, fechar, falar, puxar, trabalhar). Na captura é usado o Kinect e a biblioteca OpenNI. No processo de segmentação utiliza-se a informação do skeleton do Kinect para definir um quadro em torno da mão e segmentá-lo da imagem. Para descartar a informação do fundo é utilizada a imagem de profundidade.

Um sistema para reconhecimento automático das 26 posturas estáticas do alfabeto da língua brasileira de sinais (LIBRAS) e da língua de sinais americana (ASL) é proposto por (JUNIOR, 2014). Utiliza um sensor de RGB-D na aquisição de dados e de posse das imagens de profundidade aplica a combinação da estratégia de Casamento de Modelos, com o algoritmo Interative Closest Point (ICP) na fase de reconhecimento. O trabalho apresentou um desempenho máximo de 99,04 % de taxa de acerto no reconhecimento na ASL e 99,62 para a LIBRAS.

A criação de uma base de dados (em expansão) contendo 24 palavras em LIBRAS executadas por vários voluntários, e com planos de fundo diferentes é proposta por

(MONTEIRO et al, 2016). A extração de características é baseada em sub amostragens de imagens residuais. O trabalho apresentou uma taxa de acerto média de 75%.

2. SENSOR KINECT E KINECT SDK

Na captura da imagem foi utilizado o sensor Kinect One apresentado na Figura 2 que é formado por um laser infravermelho e um sensor CMOS monocromático, que capta os dados 3D. Uma das características que este sensor de profundidade apresenta é que ele não é sensível a variação da luz. Ele é capaz de capturar imagens tanto em ambiente bem iluminado quanto em ambiente escuro (MENDONÇA, 2013).

A resolução da câmera de profundidade é de 512x424, da câmera em RBG é de 1920x1080 (16:9), uma razão de 30 frames por segundo e resolução do microfone de 48kHz.

Figura 2: Sensor Kinect

(a) Kinect 360 (b) Kinect one

Fonte: MICROSOFT (2012)

O Kinect também possui uma câmera de cores (RGB), um sensor de profundidade que usa infravermelho e mapeia o ambiente em 3D, microfones, uma base com motor para alterar seu angulo de visão e interface USB para conexão com videogame e com o computador. A robótica vem fazendo uso deste dispositivo, devido a sua capacidade de percepção espacial.

O pacote oferece acesso as fontes de dados captadas pelos sensores: de profundidade, câmera RGB e 4 microfones do sistema de captura de áudio. Permite ainda acesso ao rastreio do esqueleto (CORREIA, 2013).

Na Tabela 1 podemos observar a diferença entre os dois sensores Kinect. A primeira é na qualidade das imagens adquiridas, na imagem RGB do ONE a resolução é full HD. Outra diferença importante está na quantidade de articulações que o novo sensor pode capturar. No sensor kinect 360 conseguimos informações sobre 20 articulações, enquanto no kinect ONE temos 25 articulações do corpo do usuário.

Tabela 1: Comparação entre o kinect V1 e V2

	V1(Xbox 360)	V2 (xbox one)
Alcance do sensor de profundidade (metros)	0.4 – 4.0	0.4 – 4.0
Resolução do canal colorido (pixel)	640x480	1920x1080
Resolução do canal de profundidade (pixel)	320x240	512x424
Canal infravermelho	None	512x424
Tipo de luz	Light coding	Tof
Canal de áudio	4-mic array 16 kHz	4-mic array 48 kHz
USB	2.0	3.0
Juntas	20	25
Campo de visão	57° H 43° V	70° H 60° V

Fonte: Adaptado de: (PAUL; BASI; NASIPURI, 2016)

Cada articulação do corpo é representada por um ponto indicado na Tabela 2. Nesta tabela foram apresentadas as 25 articulações do Kinect ONE. Para o sensor Kinect 360 até a articulação de número 20 são coincidentes com o ONE.

Tabela 2: Pontos das articulações do corpo

SpineBase	1		Knee_Left	14
SpineMid	2		Ankle_Left	15
Neckr	3		Foot_Left	16
Head	4		Hip_Right	17
Shoulder_Left	5		Knee_Right	18
Elbow_Left	6		Ankle_Right	19
Wrist_Left	7		Foot_Right	20
Hand_Left	8		SpineShoulder	21
Shoulder_Right	9		HandTip_Left	22
Elbow_Right	10		Thumb_Left	23
Wrist_Right	11		HandTip_Right	24
Hand_Right	12		Thumb_Right	25
ip_Left	13			

Fonte: Adaptado de: (MathWorks, 2018)

As 5 articulações a mais que o sensor Kinect ONE traz em seu pacote é exibida na Figura 3b. Podemos destacar os pontos 22 e 24 que representam as pontas dos dedos indicadores das mãos, os pontos 23 e 25 que representam os polegares das mãos e o ponto 21 que

representa o centro dos ombros. Estes pontos são importantes no uso do vídeo game, onde o usuário muda o comando se a mão estiver aberta ou fechada.

Figura 3: Pontos de articulação dos sensores Kinect

Fonte: Autores (2019)

3. METODOLOGIA

Para realização deste projeto foi utilizado Kinect que possui um sensor de profundidade que não é sensível a variação da luz, facilitando a captura de imagens em ambiente bem iluminados como também em ambientes com variações.

A interface do Kinect com o computador foi realizada através do software Matlab R2018a e R2015a. O Kinect tem duas entradas de vídeos, sendo 'Kinect Color Sensor' responsável pela captura das imagens nas escalas RGB e o 'Kinect Depth Sensor' para captura das imagens de profundidade, suas resoluções foram apresentadas na Tabela 1.

Na aquisição foram utilizados: Um notebook com o software Matlab 2015a com o sensor Kinect 360 e outro notebook com o Matlab 2018a com o sensor Kinect ONE. Os sensores ficaram a uma altura de aproximadamente 0,9m, os atores ficaram a uma distância do sensor de 1,5m e com um fundo a uma distância de 3,8m.

A coleta de imagens foi realizada nas cidades de São Luís e Fortaleza em dias da semana diferentes, com fundo variados. Cada ator ficava em frente aos sensores Kinects variando a rotação da mão num intervalo de tempo necessário para uma captura de pelo menos 100 imagens.

Para uma melhor diversidade no banco de dados os atores são variados em idade, sexo, cor e estaturas e são detalhados na Tabela 3. Essa diversidade foi aleatória de acordo com a disponibilidade de cada ator, não houve um estudo para escolha deles.

Tabela 3: Variações dos atores

Ator	Idade (anos)	Altura (metros)	Sexo	Cor da pele
1	19	1.60	Feminino	Parda
2	44	1.60	Masculino	Branca
3	46	1.78	Masculino	Parda
4	50	1.60	Feminino	Branca
5	25	1.55	Feminino	Branca
6	22	1.76	Masculino	Parda
7	27	1.76	Masculino	Negra

Fonte: Autores (2019)

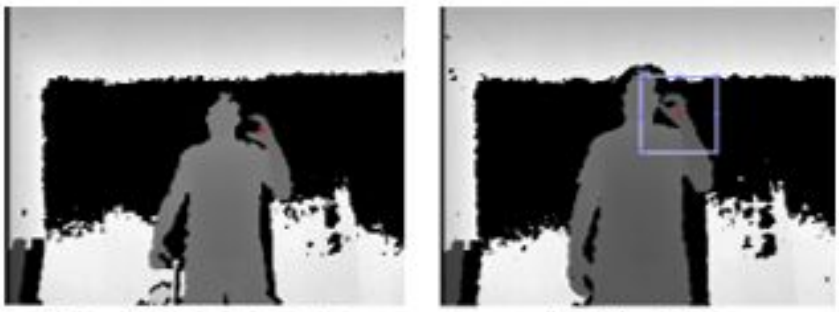
As imagens foram adquiridas no campo de visão de 70º horizontalmente e 60º verticalmente, num alcance de profundidade de 0,4m a 4,5m. Limites do sensor kinect.

No trabalho utilizamos comandos para criação de um vídeo de entrada no ambiente Matlab. Sendo que criamos dois objetos de entrada, uma para imagem em RGB e outra para a imagem de profundidade.

Foi utilizada a função "Body" do Matlab que fornece informações sobre a posição de 25 articulações do corpo do usuário. Cada articulação recebe um número e para uma melhor visualização do movimento do corpo utilizamos a função 'SkeletonConnectionMap'. Para visualizarmos o movimento do corpo através do skeleton, vamos criar um vetor com vários outros vetores, sendo que cada vetor irá fazer a ligação de uma articulação para outra, desta maneira podemos ilustrar o corpo.

Dentre estas articulações nossos estudos focam nos movimentos e acompanhamento das mãos. Esta facilidade da função em distinguir as mãos do restante do corpo em qualquer posição da imagem ajuda na determinação da região de interesse (ROI) diminuindo o tempo computacional.

Figura 4: Ponto da mão direita e ROI – Sensor 360 profundidade



(a) Ponto de articulação da mão direita (b) ROI da mão direita

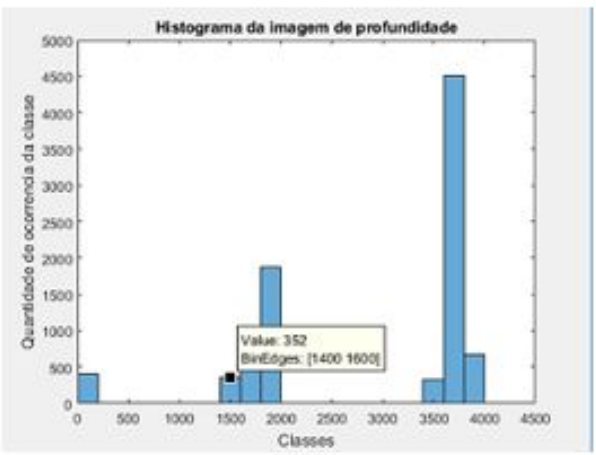
Fonte: Autores (2019)

O sensor kinect reconhece as articulações do corpo humano, para a articulação da mão temos um ponto conforme pode ser visto na Figura 4a. Ao redor do ponto que rastreia a mão direita foi criado um retângulo de forma arbitrária de 101x101 (Kinect 360) visto na Figura 4b e 161x161 (Kinect One). Os demais pontos foram omitidos pois não serão necessários nesta parte do trabalho.

Com a criação deste retângulo conseguimos definir nossa região de interesse e retirar somente a região que nos interessa, as ROI's.

No retângulo azul da Figura 4b podemos observar que temos muitas informações desnecessárias, como por exemplo o fundo da imagem. No Matlab existem vários comandos que calculam o histograma das imagens, porém com a utilização destes comandos perdemos os valores de profundidade. Para resolvermos este conflito criamos uma função que faz a leitura pixel a pixel para depois exibirmos o histograma conforme pode ser visto na Figura 5.

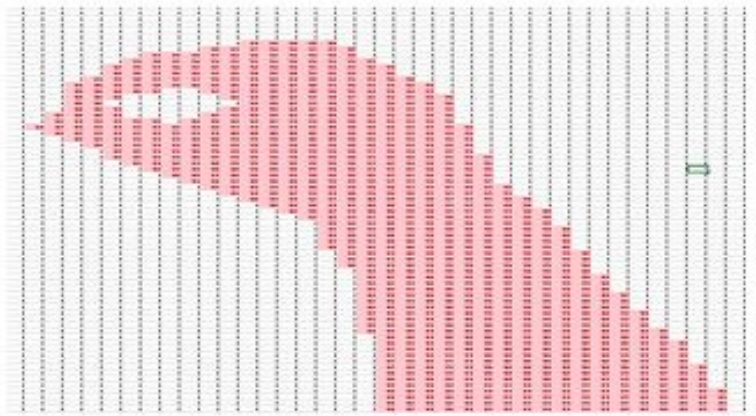
Figura 5: Histograma da imagem recortada – Kinect 360 depth



Fonte: Autores (2019)

Observamos na Figura 5 que no primeiro máximo local, após o "0", tem 352 ocorrências na intensidade de pixel de 1400 a 1600 (Essas intensidades representam as distâncias do sensor até o corpo em "mm"). Este máximo local nos fornece uma posição, no caso 1400mm, este valor acrescido de 120mm será usado como ponto de corte para separarmos a região da mão do restante do corpo e do fundo. Fazendo os valores maiores que o ponto encontrado se tonarem "0", resulta a Figura 6. Os valores são exibidos numa planilha de Excel para uma melhor visualização das distancias que foram explicados anteriormente.

Figura 6: Valores da imagem segmentada – Kinect 360 depth



Fonte: Autores (2019)

Para padronização das imagens capturadas será utilizada como base os gestos disponibilizados na página do INES (Instituto Nacional de Educação de Surdos).

Também foi utilizada a pesquisa documental e bibliográfica para informações sobre os métodos de aquisição, segmentação, segregação e extração de características de imagens.

A base de dados foi montada através de imagens capturadas pelo sensor Kinect processadas e armazenadas pelo software Matlab.

A Tabela 4 mostra a quantidade de imagens que fazem parte do banco de dados. Temos as imagens em escala RGB que foram adquiridas tanto no sensor kinect 360 como no sensor Kinect ONE. Das imagens originais realizamos a segmentação e obtivemos um novo conjunto de imagens.

Tabela 4: Quantidade de imagens por letras

	Kinect 360	Kinect One
Letras	RGB, Depth e RGB segmentada	RGB, Depth e RGB segmentada
a	596	715
b	588	723
c	629	701
d	632	713
e	646	687
f	626	715
g	621	669
i	630	759
l	627	720
m	622	722
n	659	740
o	625	673
p	638	725
q	544	624
r	518	627
s	517	649
t	525	597
u	524	631
v	517	622
Total	**11284**	**13012**

Fonte: Autores (2019)

4. RESULTADOS

4.1. KINECT 360

4.1.1. Sem Segmentação

Foram usadas imagens RGB adquiridas com o sensor Kinect360, sendo que selecionamos as imagens originais em escala RGB, totalizando 11284 imagens. Na Tabela 5 podemos visualizar a matriz de confusão onde a acurácia obtida foi de 88%. É possível observar que a letra a tem acurácia mais alta (95%) e as letras l, m e n tem as acurácias mais baixas (81%). Na acurácia da letra m temos 6% confundida com a letra n e na acurácia da letra n temos 7% confundida com a letra m.

Tabela 5: Matriz de confusão das imagens - 360 RGB

Letras	a	b	c	d	e	f	G	i	L	M	n	o	p	q	R	s	t	u	v
a	0,95	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,01	0,00	0,00	0,00	0,00
b	0,00	0,91	0,00	0,01	0,01	0,00	0,00	0,01	0,00	0,01	0,00	0,00	0,01	0,00	0,01	0,01	0,00	0,01	0,01
c	0,01	0,00	0,87	0,02	0,02	0,00	0,00	0,00	0,01	0,01	0,01	0,02	0,01	0,00	0,00	0,00	0,00	0,00	0,00
d	0,00	0,01	0,01	0,84	0,01	0,02	0,01	0,02	0,01	0,01	0,01	0,03	0,00	0,01	0,01	0,01	0,00	0,01	0,00
e	0,01	0,02	0,00	0,01	0,88	0,01	0,02	0,01	0,00	0,01	0,00	0,00	0,01	0,00	0,00	0,01	0,00	0,00	0,00
f	0,00	0,00	0,01	0,01	0,00	0,86	0,03	0,01	0,00	0,01	0,01	0,01	0,00	0,00	0,00	0,00	0,04	0,01	0,00
g	0,01	0,01	0,00	0,00	0,01	0,01	0,88	0,02	0,02	0,00	0,01	0,00	0,00	0,00	0,00	0,01	0,01	0,00	0,00
i	0,00	0,01	0,00	0,01	0,01	0,01	0,03	0,85	0,02	0,00	0,01	0,00	0,00	0,01	0,00	0,01	0,01	0,01	0,00
l	0,00	0,00	0,01	0,01	0,01	0,01	0,02	0,03	0,81	0,00	0,02	0,00	0,02	0,01	0,00	0,01	0,02	0,01	0,02
m	0,01	0,00	0,01	0,00	0,01	0,00	0,00	0,01	0,00	0,81	0,06	0,01	0,00	0,04	0,00	0,01	0,00	0,01	0,01
n	0,01	0,00	0,01	0,00	0,00	0,00	0,01	0,00	0,01	0,07	0,81	0,01	0,02	0,03	0,00	0,01	0,00	0,01	0,00
o	0,01	0,00	0,01	0,02	0,01	0,01	0,00	0,01	0,01	0,00	0,01	0,88	0,01	0,01	0,00	0,01	0,00	0,00	0,00
p	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,00	0,01	0,01	0,01	0,01	0,93	0,01	0,00	0,00	0,00	0,00	0,00
q	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,00	0,00	0,01	0,01	0,00	0,01	0,93	0,00	0,00	0,01	0,00	0,00
r	0,00	0,01	0,00	0,01	0,00	0,00	0,02	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,89	0,01	0,00	0,02	0,01
s	0,00	0,00	0,00	0,00	0,01	0,00	0,01	0,00	0,01	0,00	0,01	0,01	0,00	0,00	0,01	0,93	0,01	0,00	0,00
t	0,00	0,00	0,00	0,00	0,00	0,02	0,00	0,01	0,01	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,92	0,01	0,00
u	0,00	0,01	0,00	0,00	0,00	0,01	0,01	0,01	0,01	0,01	0,01	0,00	0,00	0,00	0,05	0,01	0,01	0,77	0,08
v	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,01	0,01	0,00	0,00	0,00	0,00	0,00	0,02	0,00	0,00	0,03	0,91
Acurácia = 0,88																			

Fonte: Autores (2019)

4.1.2. Com Segmentação da Imagem RGB Usando a Imagem de Profundidade Segmentada

Para melhorarmos a acurácia do nosso classificador realizamos a seguinte segmentação. Utilizando a imagem de profundidade (Depth) aplicando metodologia descrita anteriormente conseguimos separar a mão do restante da imagem obtendo a Figura 7b. Esta imagem segmentada é multiplicada por cada componente da imagem RGB (R, G e B) 7a resultando na imagem segmentada Figura 7c.

Figura 7: Imagens RGB segmentadas utilizando imagem de profundidade segmentada

(a) Imagem Original RGB (b) Imagem profundidade segmentada (c) Imagem RBG segmentada

Fonte: Autores (2019)

Com este novo conjunto de imagens RGB segmentadas aplicamos o classificador e obtivemos uma nova matriz de confusão. Podemos observar que com as imagens segmentadas obtivemos uma melhora no nosso classificador. Sua acurácia foi 94% conforme pode ser visto na Tabela 6.

Tabela 6: Matriz de confusão das imagens – 360 RGB

Letras	a	b	c	d	e	F	g	i	l	M	n	o	p	q	R	s	t	u	v
a	0,95	0,00	0,00	0,00	0,01	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,02	0,00	0,00	0,00
b	0,00	0,97	0,00	0,01	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,00	0,00
c	0,00	0,00	0,96	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,01	0,00	0,00	0,00	0,00	0,00	0,00
d	0,00	0,00	0,01	0,87	0,02	0,00	0,01	0,01	0,00	0,00	0,00	0,05	0,00	0,00	0,02	0,01	0,00	0,00	0,00
e	0,01	0,01	0,00	0,00	0,93	0,00	0,01	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,02	0,00	0,00	0,00
f	0,00	0,01	0,00	0,00	0,00	0,87	0,02	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,01	0,06	0,00	0,00
g	0,02	0,00	0,00	0,00	0,00	0,00	0,93	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00
i	0,01	0,00	0,00	0,00	0,01	0,00	0,00	0,96	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,00
l	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,98	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01
m	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,81	0,15	0,00	0,00	0,02	0,00	0,00	0,00	0,00	0,00
n	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,13	0,85	0,00	0,00	0,01	0,00	0,00	0,00	0,00	0,00
o	0,00	0,00	0,01	0,02	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,95	0,00	0,00	0,00	0,00	0,00	0,00	0,00
p	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,99	0,00	0,00	0,00	0,00	0,00	0,00
q	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,99	0,00	0,00	0,00	0,00	0,00
r	0,00	0,00	0,00	0,01	0,00	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,93	0,00	0,00	0,03	0,00
s	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,97	0,00	0,00	0,00
t	0,00	0,00	0,00	0,00	0,00	0,03	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,95	0,00	0,00
u	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,04	0,00	0,01	0,91	0,01
v	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,98
Acurácia = 0,94																			

Fonte: Autores (2019)

4.2. KINECT ONE

4.2.1. Sem Segmentação

Utilizando as imagens adquiridas com o sensor Kinect One aplicamos o classificador obtendo a matriz de confusão Tabela 7. Nesta matriz a acurácia foi de 89%. As letras a e b apresentaram maior acurácia (98%) e a letra n teve a menor acurácia (77%).

Tabela 7: matriz de confusão das imagens RGB One

Letras	a	b	c	d	E	F	g	i	l	M	n	o	p	q	R	s	t	u	v
a	0,98	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
b	0,01	0,98	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
c	0,01	0,00	0,88	0,02	0,01	0,01	0,01	0,00	0,00	0,00	0,01	0,03	0,01	0,00	0,00	0,01	0,01	0,00	0,00
d	0,01	0,00	0,02	0,88	0,01	0,01	0,01	0,01	0,00	0,00	0,00	0,02	0,01	0,00	0,01	0,01	0,02	0,00	0,00
e	0,00	0,00	0,01	0,01	0,91	0,00	0,01	0,01	0,01	0,00	0,01	0,00	0,01	0,00	0,01	0,01	0,01	0,00	0,01
f	0,00	0,01	0,00	0,00	0,01	0,88	0,00	0,02	0,00	0,01	0,01	0,00	0,01	0,00	0,01	0,01	0,04	0,00	0,00
g	0,02	0,01	0,01	0,01	0,02	0,01	0,84	0,01	0,03	0,01	0,00	0,00	0,00	0,00	0,01	0,00	0,01	0,01	0,01
i	0,00	0,01	0,01	0,01	0,01	0,01	0,00	0,92	0,01	0,00	0,00	0,01	0,00	0,00	0,00	0,01	0,00	0,00	0,00
l	0,01	0,01	0,01	0,01	0,01	0,01	0,03	0,01	0,88	0,00	0,00	0,01	0,00	0,01	0,00	0,02	0,01	0,01	0,00
m	0,00	0,01	0,00	0,01	0,01	0,01	0,00	0,00	0,00	0,87	0,03	0,00	0,00	0,03	0,01	0,00	0,01	0,00	0,01
n	0,01	0,00	0,00	0,00	0,01	0,02	0,00	0,01	0,01	0,10	0,77	0,00	0,01	0,03	0,00	0,01	0,00	0,01	0,00
o	0,00	0,00	0,01	0,02	0,00	0,01	0,00	0,01	0,01	0,00	0,00	0,92	0,01	0,01	0,00	0,00	0,00	0,01	0,01
p	0,00	0,00	0,00	0,01	0,00	0,01	0,00	0,00	0,00	0,01	0,00	0,01	0,93	0,01	0,00	0,00	0,00	0,01	0,00
q	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,00	0,01	0,02	0,00	0,01	0,94	0,01	0,00	0,00	0,00	0,00
r	0,00	0,01	0,00	0,01	0,01	0,01	0,01	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,86	0,01	0,02	0,03	0,02
s	0,01	0,00	0,01	0,00	0,02	0,01	0,01	0,01	0,02	0,00	0,01	0,01	0,01	0,02	0,04	0,81	0,01	0,01	0,00
t	0,00	0,01	0,01	0,01	0,01	0,02	0,00	0,01	0,01	0,00	0,00	0,01	0,01	0,01	0,01	0,00	0,86	0,01	0,01
u	0,01	0,01	0,00	0,01	0,00	0,00	0,01	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,02	0,02	0,02	0,87	0,04
v	0,00	0,00	0,00	0,01	0,01	0,01	0,01	0,01	0,00	0,00	0,00	0,01	0,00	0,00	0,01	0,01	0,00	0,03	0,89
Acurácia = 0,89																			

Fonte: Autores (2019)

4.2.2. Com Segmentação da Imagem RGB Usando a Imagem de Profundidade Segmentada

Para melhorarmos a acurácia do classificador não podemos aplicar diretamente a metodologia utilizada com o sensor Kinect 360, devido a resoluções diferentes entre as imagens em RGB e profundidade, descritas na Tabela 1. Na Figura 8 podemos observar que os retângulos que determinam as ROI's, apesar de possuírem as mesmas dimensões (161x161), devido as resoluções diferentes a ROI captura regiões distintas do mesmo ator. No caso da imagem profundidade, esta capturou a cabeça do ator, enquanto a RGB apenas uma parte da cabeça.

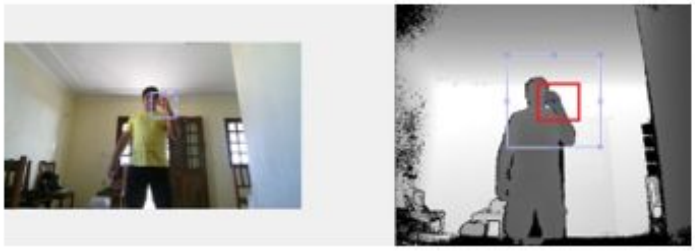
Figura 8: Imagens One RGB e profundidade

Fonte: Autores (2019)

Nosso ponto de partida é a imagem RGB, devemos criar uma forma de adquirir informações da imagem profundidade, devido esta ter o mapa de profundidade, o que nos ajuda na segmentação.

Observando a Figura 9, temos a necessidade de cortar parte da imagem profundidade (retângulo vermelho) de modo que as regiões sejam compatíveis entre as duas imagens.

Figura 9: Imagens One RGB e profundidade com retângulo a ser recortado



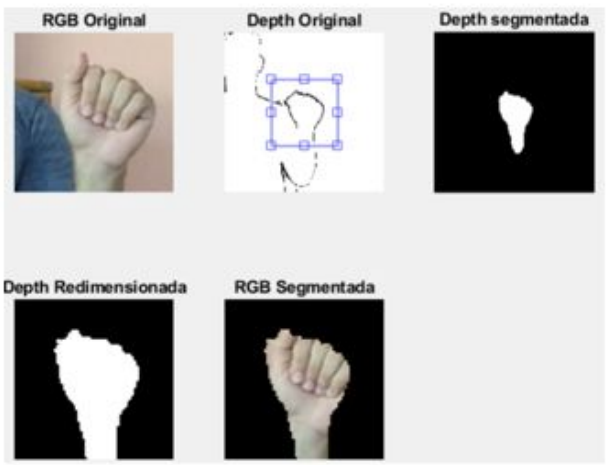
Fonte: Autores (2019)

Na imagem de profundidade serão aplicados os seguintes processos, respectivamente:

- Aplicar na imagem profundidade Original a metodologia com base no histograma e obtemos a imagem profundidade segmentada;
- Na imagem profundidade segmentada recortamos a região determinada pela área de corte, porém esta imagem contém resoluções diferentes da RGB Original.
- Para resolvermos este problema é necessário aplicar um redimensionamento na imagem recortada para a mesma resolução da imagem RGB.

Estes passos realizados nas imagens descritas acima são demonstrados na Figura 10.

Figura 10: Imagens testadas para segmentação



Fonte: Autores (2019)

Para encontramos um valor de área de corte da imagem profundidade foram testados diversos valores, sendo encontrado o valor 67x67 como aceitável.

A Figura 10 mostra como a validação do valor foi realizada. Após processo de segmentação aplicamos esse novo conjunto de imagens no classificador e obtivemos a matriz de confusão Tabela 8 onde a acurácia foi de 92%.

Tabela 8: matriz de confusão das imagens - RGB One

Letras	a	b	c	d	e	f	g	i	L	M	N	o	p	q	r	s	t	u	v
a	0,96	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,00
b	0,00	0,95	0,00	0,00	0,01	0,00	0,00	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,02	0,00
c	0,00	0,00	0,93	0,00	0,01	0,01	0,00	0,01	0,00	0,00	0,00	0,02	0,00	0,00	0,00	0,01	0,01	0,00	0,00
d	0,00	0,00	0,00	0,93	0,00	0,00	0,00	0,02	0,00	0,00	0,00	0,02	0,00	0,00	0,02	0,01	0,00	0,00	0,00
e	0,01	0,01	0,00	0,01	0,92	0,00	0,00	0,01	0,00	0,00	0,00	0,01	0,00	0,00	0,01	0,02	0,00	0,00	0,00
f	0,00	0,01	0,00	0,01	0,01	0,81	0,01	0,02	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,12	0,00	0,01
g	0,01	0,00	0,00	0,00	0,00	0,00	0,95	0,01	0,01	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,01	0,00	0,00
i	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,96	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,00	0,01
l	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,98	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
m	0,01	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,84	0,13	0,00	0,00	0,02	0,00	0,00	0,00	0,00	0,00
n	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,16	0,77	0,00	0,00	0,04	0,00	0,01	0,00	0,00	0,00
o	0,00	0,00	0,03	0,01	0,01	0,00	0,00	0,01	0,00	0,00	0,00	0,91	0,00	0,00	0,00	0,02	0,00	0,00	0,00
p	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,00	0,00	0,99	0,00	0,00	0,00	0,00	0,00	0,00
q	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,02	0,00	0,00	0,96	0,00	0,00	0,00	0,00	0,00
r	0,00	0,00	0,00	0,01	0,01	0,00	0,02	0,01	0,00	0,00	0,01	0,00	0,00	0,00	0,88	0,00	0,00	0,05	0,01
s	0,01	0,01	0,01	0,00	0,02	0,00	0,01	0,01	0,00	0,00	0,00	0,02	0,00	0,00	0,01	0,90	0,00	0,01	0,00
t	0,00	0,01	0,00	0,01	0,00	0,03	0,00	0,02	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,93	0,00	0,01
u	0,00	0,01	0,00	0,02	0,00	0,01	0,00	0,01	0,00	0,00	0,00	0,01	0,00	0,00	0,08	0,01	0,01	0,84	0,02
v	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,02	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,00	0,01	0,96
Acurácia = 0,92																			

Fonte: Autores

Na Tabela 9 apresentamos o desempenho dos dois sensores Kinect com suas acurácias. Podemos verificar que o sensor Kinect One já apresenta uma melhor acurácia nas imagens não segmentadas, isto se deve ao fato que sua resolução é melhor que o Kinect 360, conforme em tabela comparativa entre os dois sensores.

Tabela 9: Acurácia das imagens com Kinect 360 e One

	Não segmentada Acurácia (%)	Segmentada Acurácia (%)
Kinect 360	88	94
Kinect One	89	92

Fonte: Autores (2019)

5. CONCLUSÃO

O reconhecimento de gestos tem sido um tema abordado por diversos pesquisadores. A interface homem máquina ainda carece de boas ferramentas para implementação. O reconhecimento de linguagem de sinais LIBRAS vem sendo estudado e desenvolvidas técnicas obtidas em imagem sem RGB com fundos uniformes para facilitar o processo de segmentação. Neste trabalho foi apresentado na forma de matriz de confusão que um fundo não uniforme pode ter uma variação muito grande no resultado.

Apresentamos neste trabalho uma avaliação das possíveis soluções das etapas de segmentação. As imagens RGB's capturadas com o sensor Kinect 360 foram segmentadas a partir de suas imagens de profundidade pelos seus respectivos histogramas. Essa segmentação apresentou uma melhora na acurácia de 6%.

Nas imagens RGB capturadas pelo sensor Kinect ONE utilizamos a mesma técnica de segmentação usada no Kinect 360, porém seus resultados foram inferiores aos valores obtidos nas imagens sem segmentação. A solução encontrada para resolução do problema descrita na seção dos resultados atendeu de forma satisfatória, onde sua acurácia teve uma melhora de 3%.

REFERÊNCIAS

ALVARENGA, Matheus L. T.; CORREA, Diogo S. O.; OSÓRIO., Fernando S. **Redes neurais artificiais no reconhecimento de gestos usando o kinect**. Computer on the beach, 2012, Itajaí - SC. Disponível em: https://siaiap32.univali.br/seer/index.php/acotb/article/view/6602/3747. Acesso em: 03 jun. 2016.

ANJO, Mauro dos S. **Avaliação das técnicas de segmentação, modelagem e classificação para o reconhecimento automático de gestos e proposta de uma solução para classificar gestos de libra em tempo real**. Dissertação (Ciência da computação) — Universidade Federal de São Carlos, 2013. Disponível em: https://repositorio.ufscar.br/handle/ufscar/523. Acesso em: 26 ago. 2016.

CORREIA, Miguel M. **Reconhecimento de elementos da língua gestual portuguesa com Kinect**. Dissertação (Engenharia de Eletrotécnica e de Computadores) — Universidade do Porto, 2013. Disponível em: http://hdl.handle.net/10216/68032. Acesso em: 10 abr. 2017.

FILHO, Ogê. M.; NETO., Hugo. V. **Processamento digital de imagens**. [S.l.]: Brasport, 1999.

JUNIOR., Juarez. P. da S. **Alinhamento de imagens de profundidade com aplicação no reconhecimento da língua de sinais**. Dissertação (Informática) — Universidade de Brasília, 2014. Disponível em: http://repositorio.unb.br/handle/10482/16978. Acesso em: 26 ago. 2016

MATHWORKS, **Acquire Image and Body Data Using Kinect V2**, 2018, Disponível em: https://www.mathworks.com/help/supportpkg/kinectforwindowsruntime/ug/detect-the-kinect-v2-devices.html Acesso em: 22 mai. 2018.

MENDONÇA., Vinícius. G. de. **Método para classificação de um conjunto de gestos usando Kinect**. Dissertação (Mestrado em informática) — Pontifícia Universidade Católica do Paraná, 2013. Disponível em: https://www.ppgia.pucpr.br/pt/arquivos/mestrado/dissertacoes/2013/vinicius-godoy-VF.pdf. Acesso em: 26 ago. 2016.

MICROSOFT, R. D. S. **Kinect Sensor**. 2012 Disponível em: https://docs.microsoft.com/en-us/previous-versions/microsoft robotics/hh438998(v=msdn.10)?redirectedfrom=MSDN. Acesso em: 10 mai. 2017.

MONTEIRO, Carlos H. de A. et al. **Um sistema de baixo custo para reconhecimento de gestos em libras utilizando visão computacional**. SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES, XXXIV, 2016, Santarém - PA. Disponível em: http://www.sbrt.org.br/sbrt2016/anais/ ST11/1570279225.pdf. Acesso em: 09 nov. 2016.

PAUL, S.; BASU, S.; NASIPURI, M. **Microsoft Kinect in gesture recognition:** A short review. [S.l.: s.n.], 2016.

PAVAN, Adilson R.; CAZHURRIRO, Jaime; MODESTO, Fabio. **Reconhecimento de gestos com segmentação de imagens dinâmicas aplicadas a libras**. Anuário da produção de iniciação científica discente, v. 13, n. 20, 2010. Disponível em: http: //repositorio.pgsskroton.com.br/bitstream/123456789/1240/1/artigo%2023.pdf. Acesso em: 10 abr. 2017.

QIDWAI, Uvais.; CHEN, C.-h. **Digital image processing:** an algorithmic approach with MATLAB. [S.l.]: Chapman and Hall/CRC, 2009. Unico. ISBN 1138115185.

SOARES, Thiago B. de M. M. J.; RAIA, Fábio. **Utilizando o kinect como auxílio sensorial para portadores de deficiências visuais.** COBENGE, XLII, 2014, Juiz de Fora - MG. Disponível em: http://198.136.59.239/~abengeorg/cobenge-2014/Artigos/129269.pdf. Acesso em: 03 jun. 2016.

@revistaeai

revistaeducacaoarteinclusao
@gmail.com

revista eai
educação,
artes &
inclusão