



Visão Computacional na Agropecuária: Oportunidades e Desafios

Prof. Dr. Hemerson Pistori

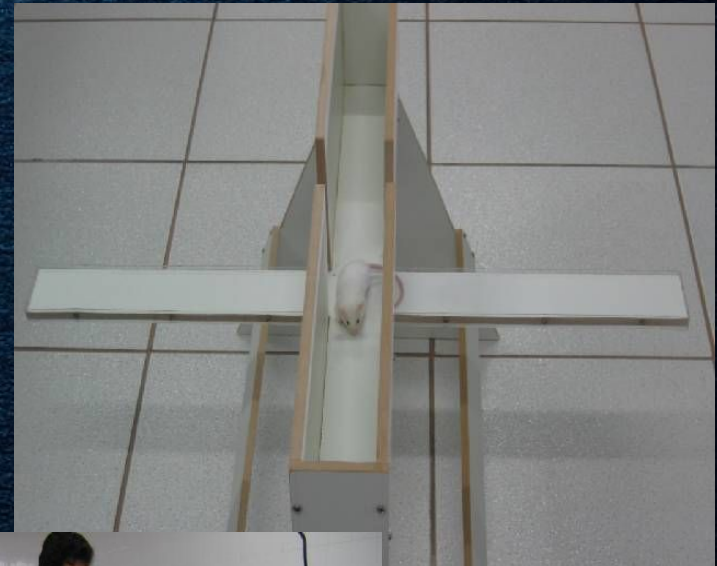
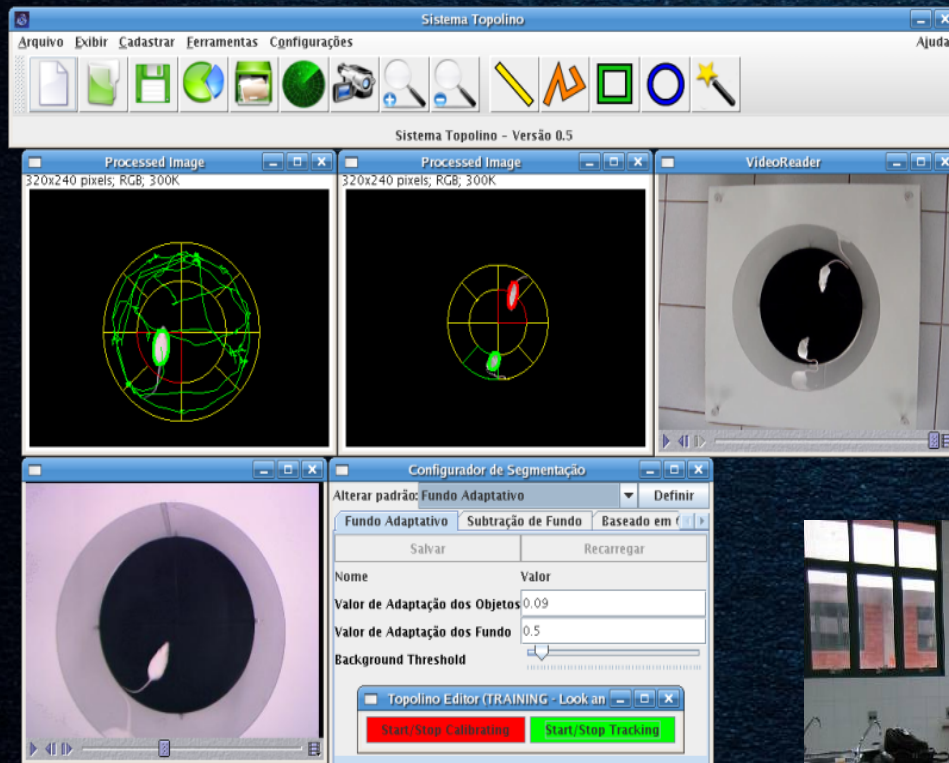
Grupo de Pesquisa Inovisão



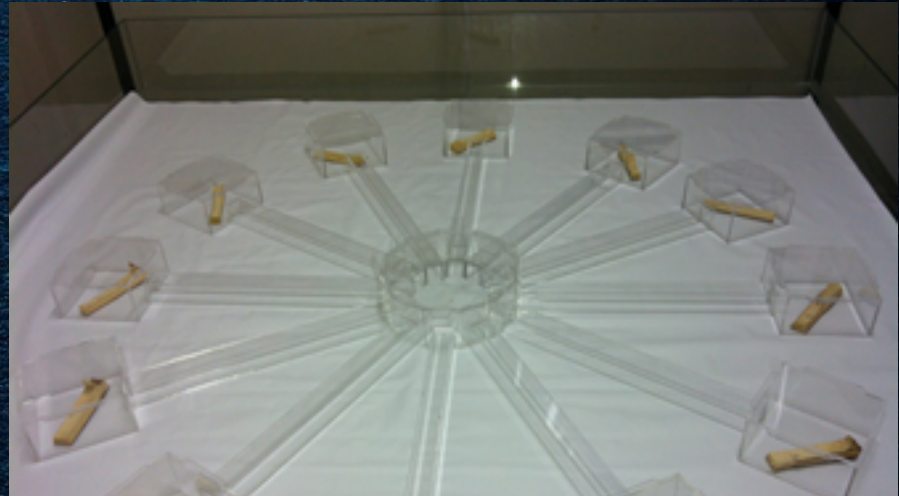
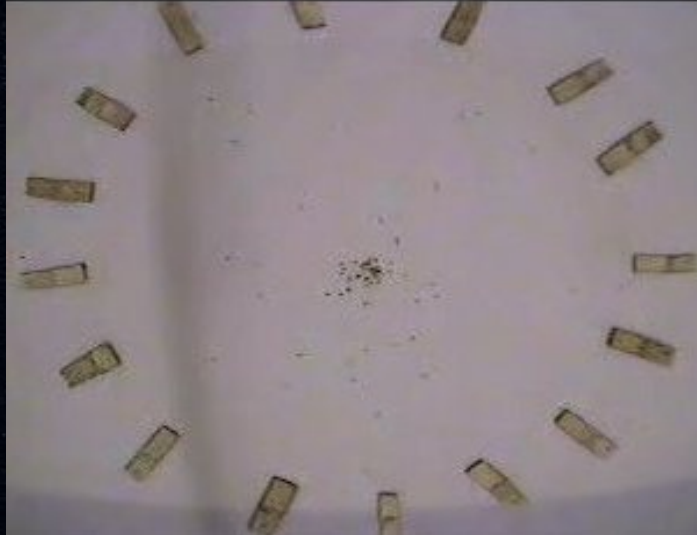
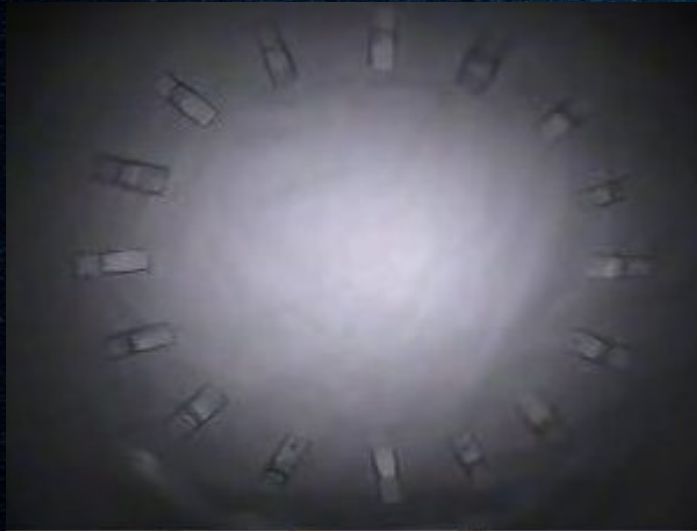
*Universidade Católica Dom Bosco
Universidade Federal do Mato Grosso do Sul
Campo Grande, MS, Brasil*



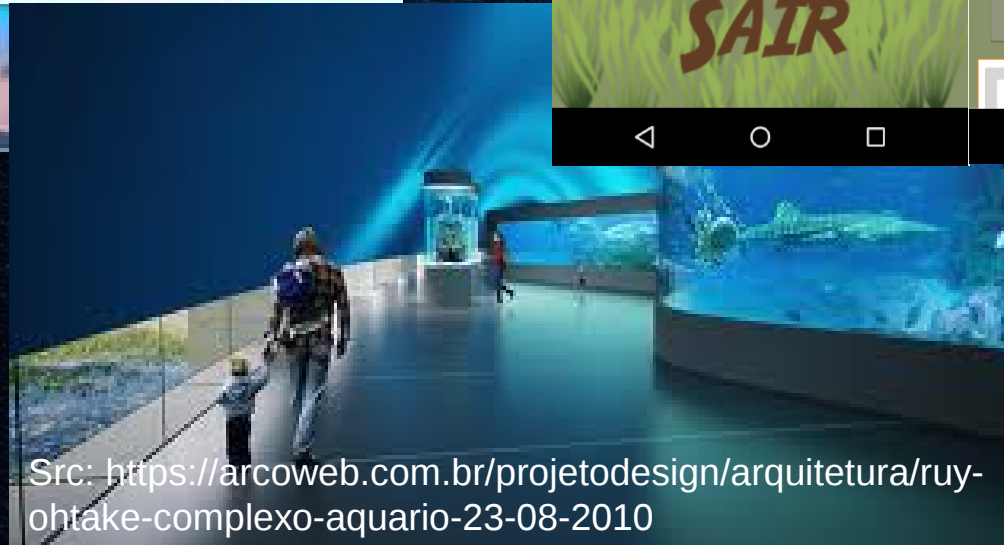
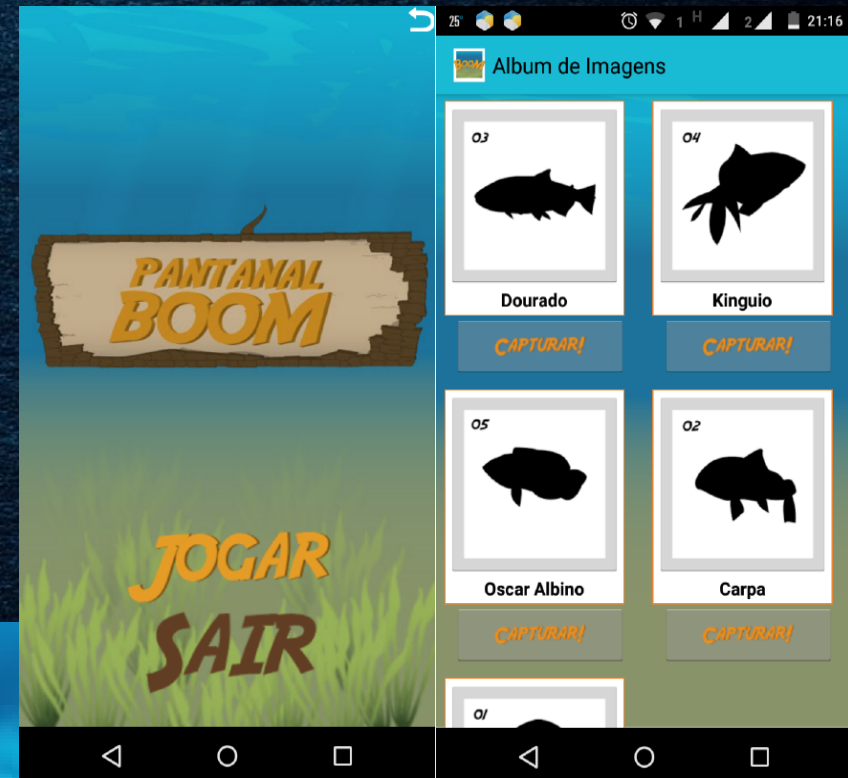
Topolino – Comportamento de Camundongos com VC



Hábito Alimentar de Carunchos do Bambu

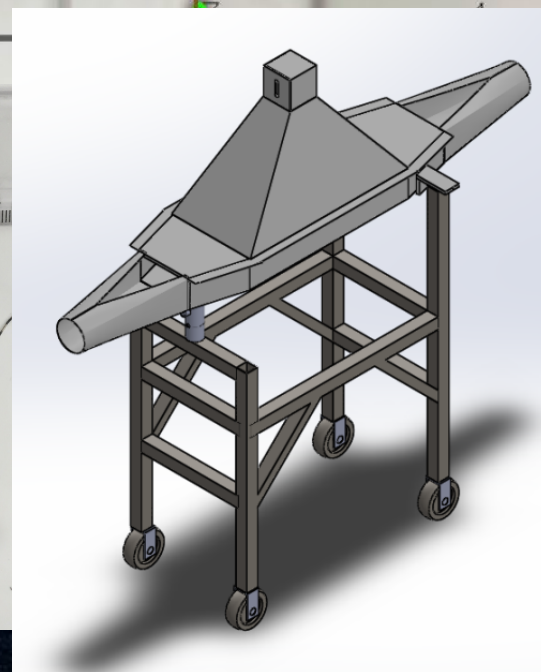
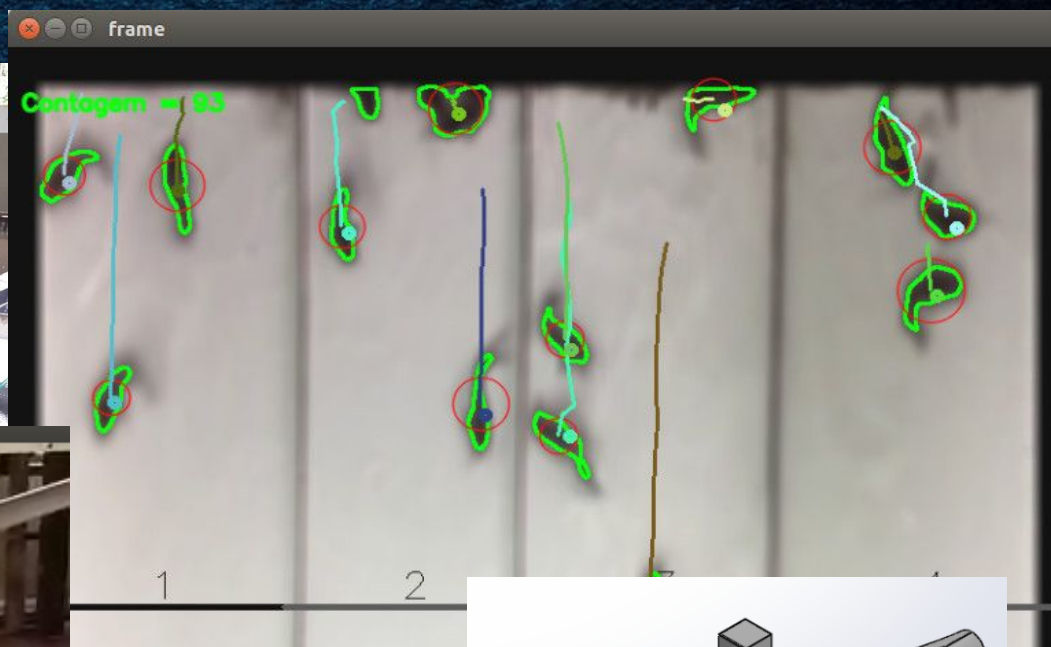


Reconhecimento de Espécies de Peixe

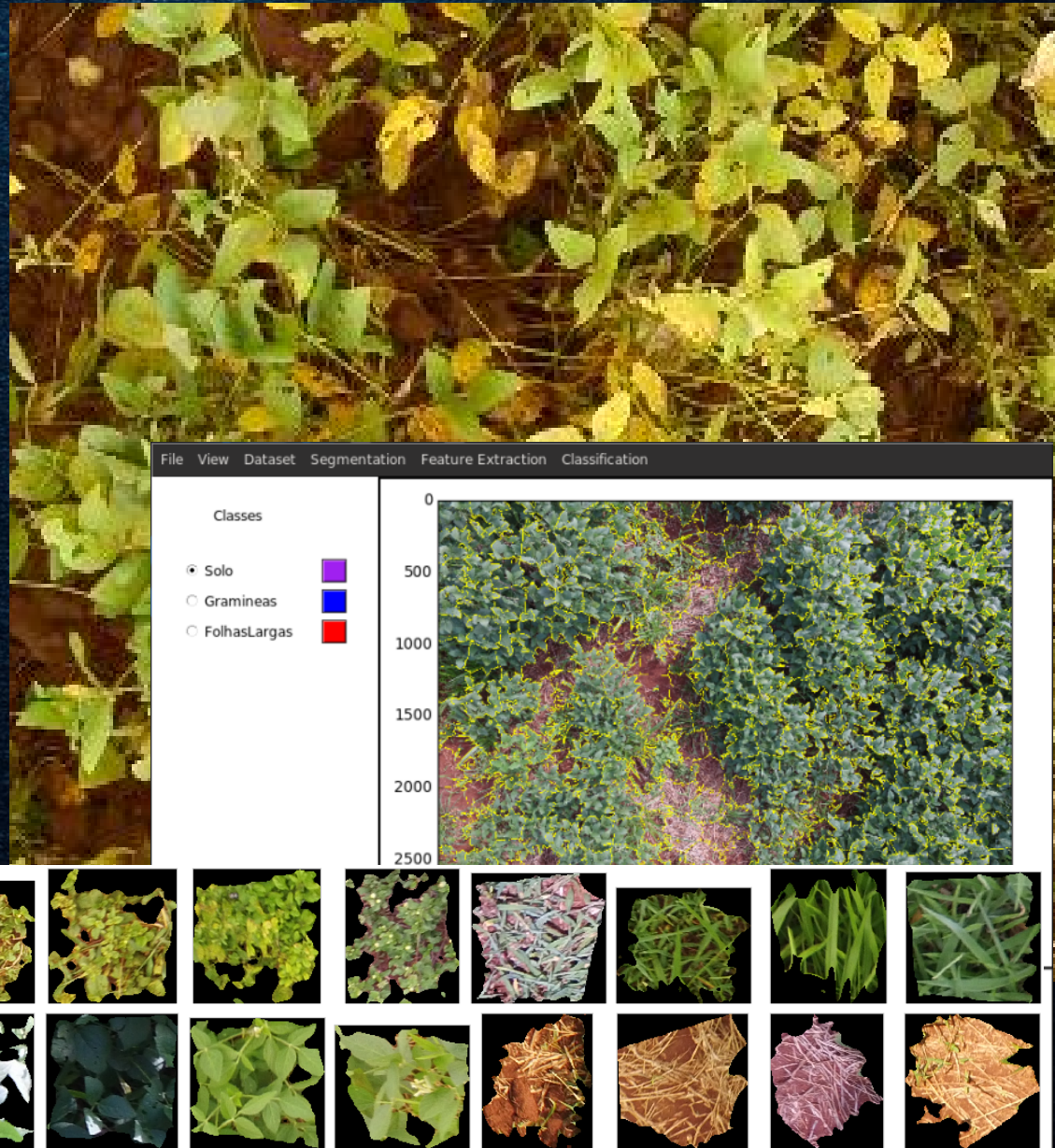
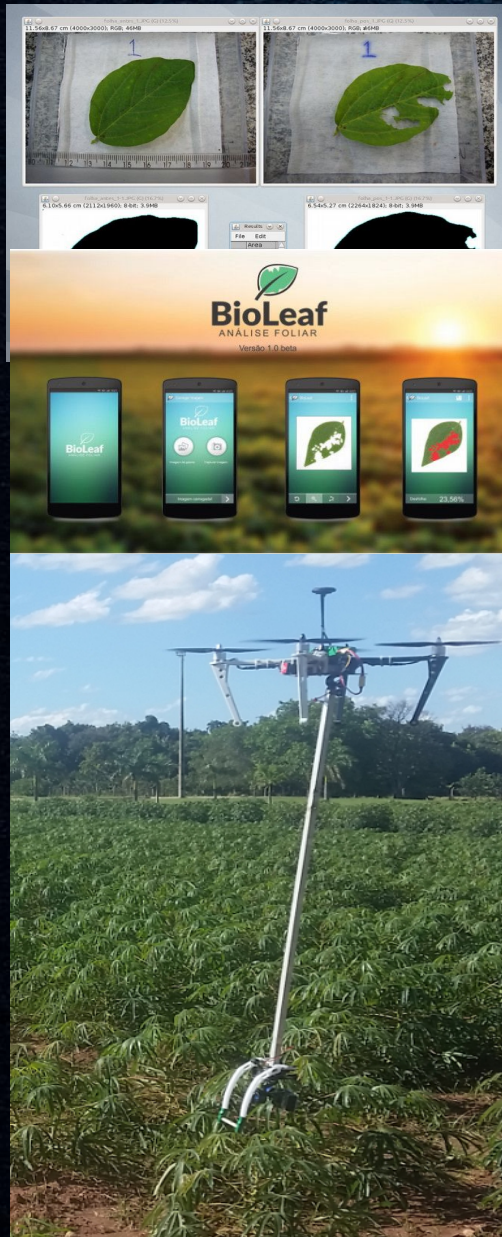


Src: <https://arcoweb.com.br/projetodesign/arquitetura/ruy-ohtake-complexo-aquario-23-08-2010>

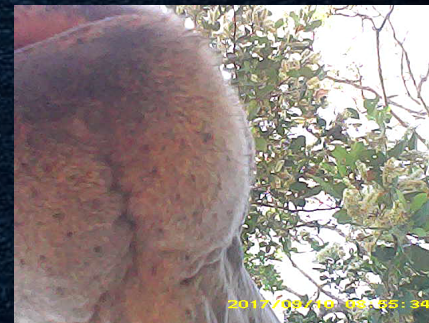
Contagem e Estimativa de Massa de Alevinos



Agricultura de Precisão e Visão Computacional



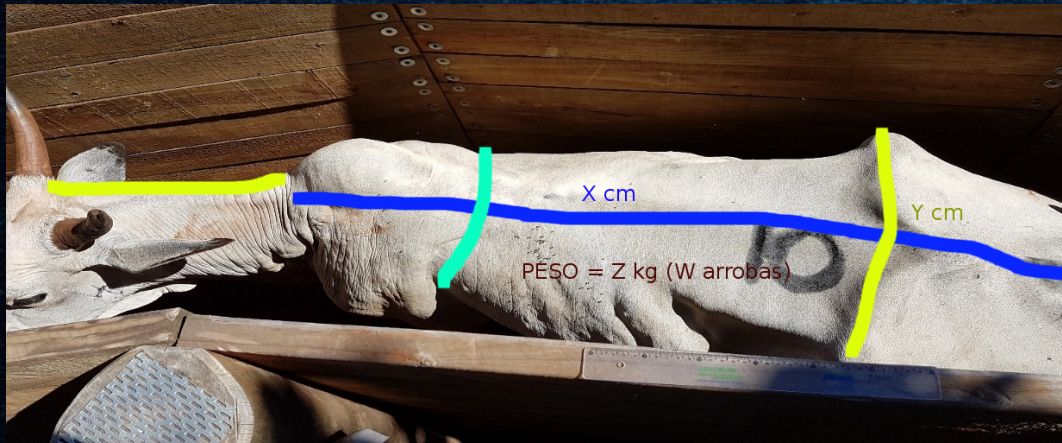
Pecuária de Precisão e Visão Computacional



Canga para VC



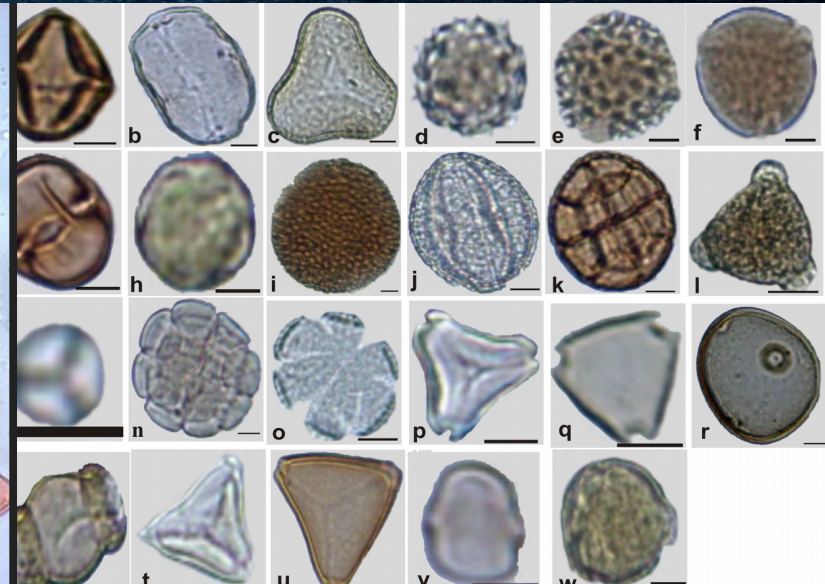
Pecuária de Precisão e Visão Computacional



Pecuária de Precisão e Visão Computacional



Palinologia Forense e Visão Computacional

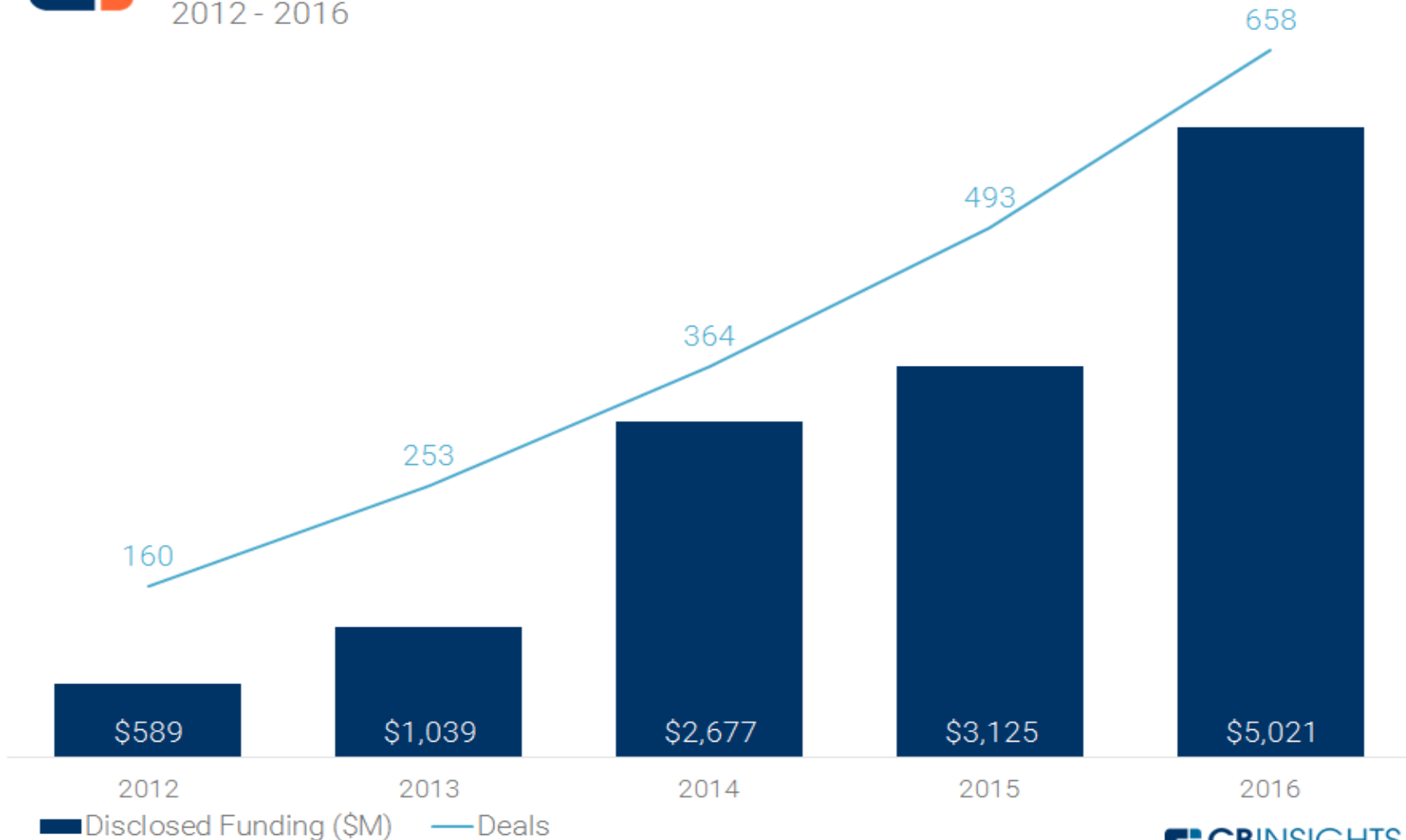


Evolução Investimentos – 2011 - 2016



AI ANNUAL GLOBAL FINANCING HISTORY

2012 - 2016



Pesquisa de Opinião entre Especialistas em IA

TOP100 autores mais citados
Microsoft Academic

ALL = TOP100+
Participantes Eventos

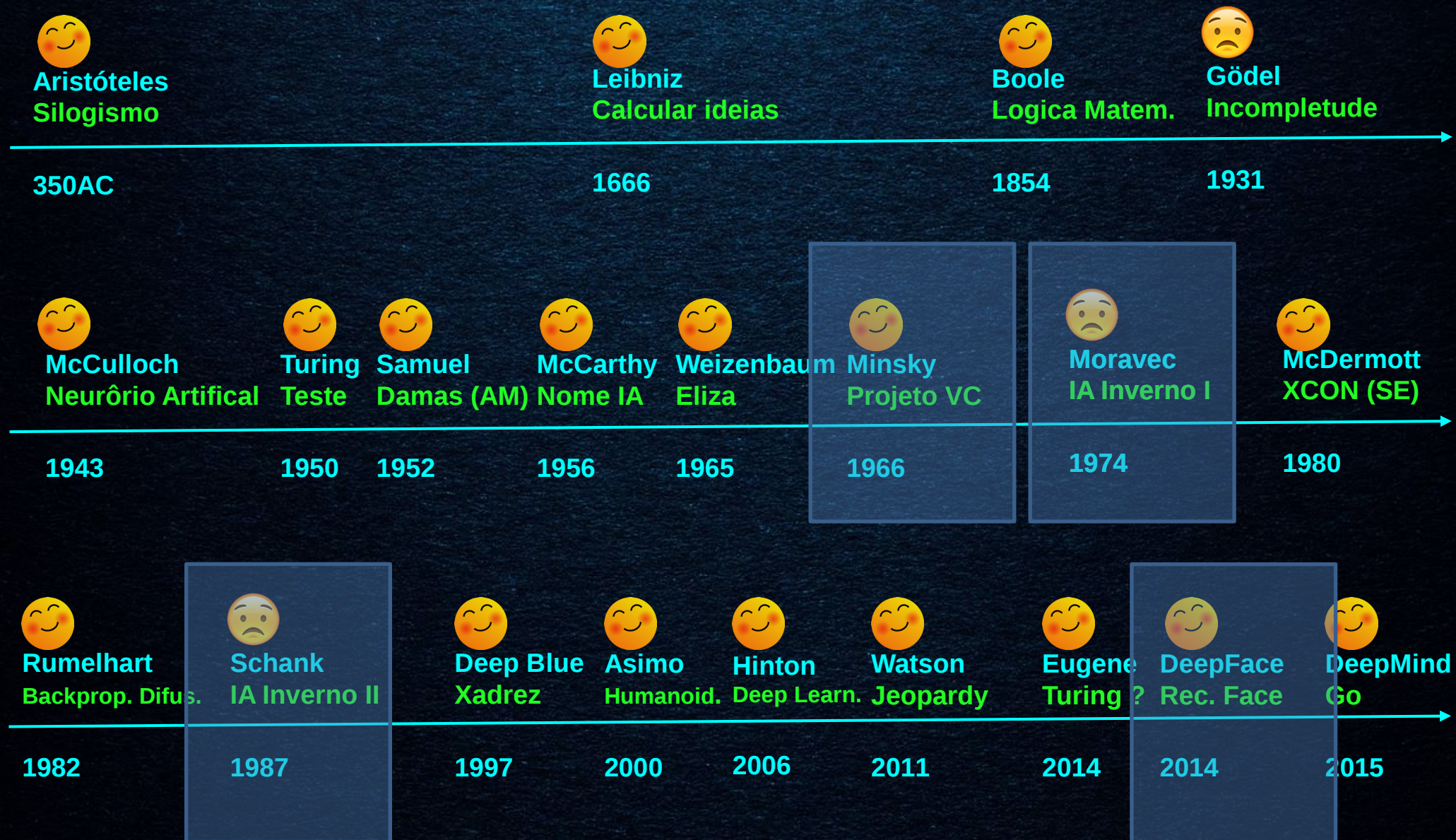
TOP100	Median	Mean	St. Dev.
10%	2024	2034	33
50%	2050	2072	110
90%:	2070	2168	342
ALL	Median	Mean	St. Dev.
10%:	2022	2036	59
50%:	2040	2081	153
90%:	2075	2183	396

Future Progress in Artificial Intelligence: A Survey of Expert Opinion

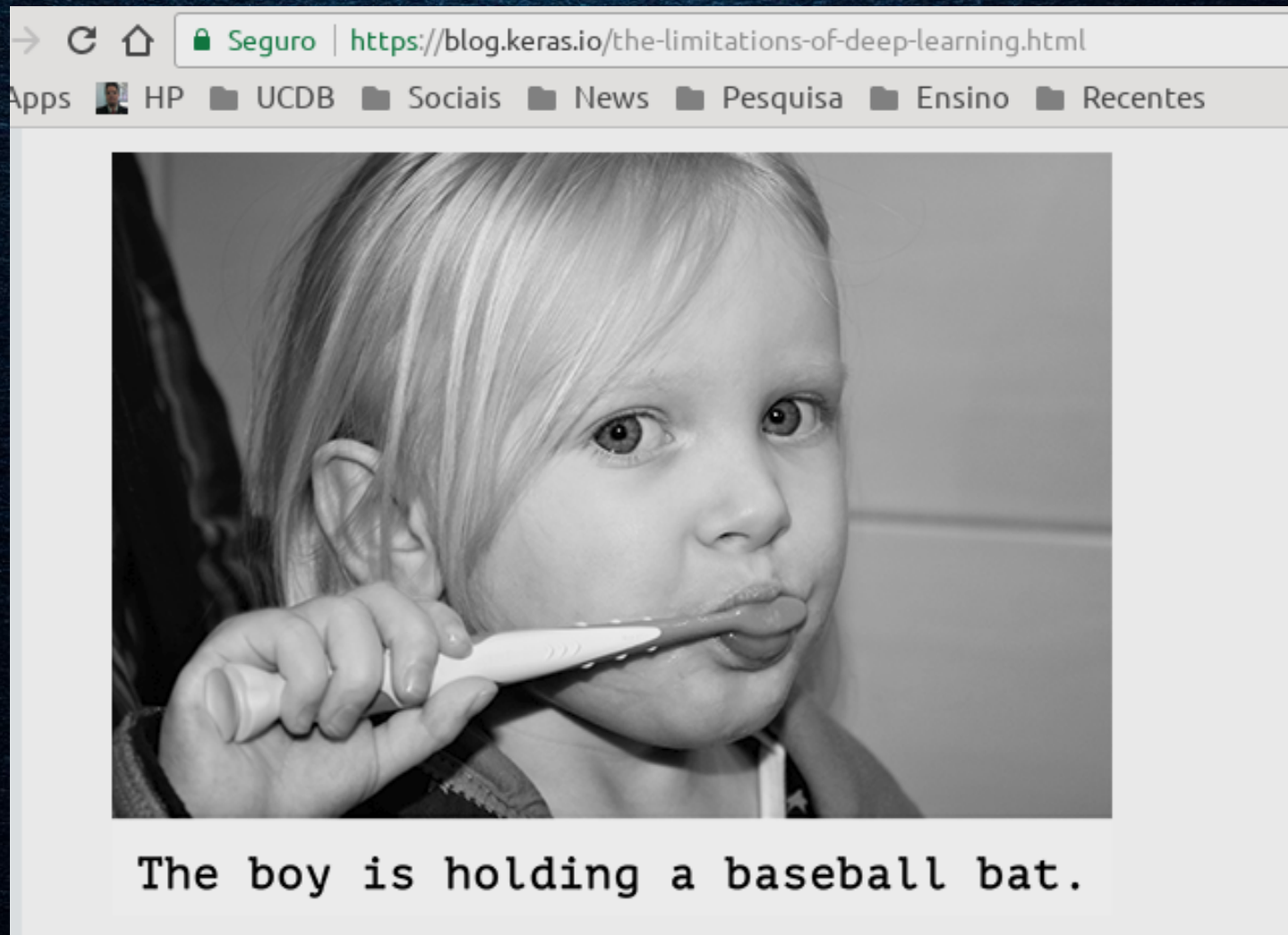
Vincent C. Müller ^{a,b} & Nick Bostrom ^a

^{a)} Future of Humanity Institute, Department of Philosophy & Oxford Martin School,
University of Oxford. ^{b)} Anatolia College/ACT, Thessaloniki

Marcos Históricos – Inteligência Artificial e Visão Computacional



Limitações das Técnicas mais Utilizadas Atualmente



O garoto está segurando um taco de baseball

Limitações das Técnicas mais Utilizadas Atualmente

TEMPE POLICE DEPARTMENT / AFP



 www.presstv.ir

 facebook.com/presstv

 twitter.com/presstv



Limitações das Técnicas mais Utilizadas Atualmente

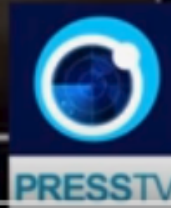
TEMPE POLICE DEPARTMENT / AFP



 www.presstv.ir

 facebook.com/presstv

 twitter.com/presstv



Reconhecimento Sintático de Padrões

```
#include <opencv2/imgproc/imgproc.hpp>
#include <sys/stat.h>
#include <pthread.h>
#include <errno.h>

using namespace cv;
using namespace std;

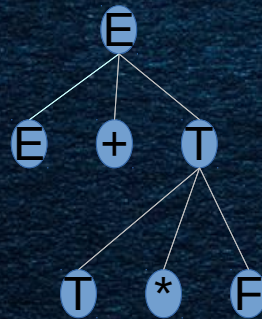
const char configFileNm[] = "../data/config.xml";

string descriptorType = "SURF";
string matcherType = "FlannBased";

void detectKeypoints( const Mat& queryImage, vector<KeyPo
queryKeypoints, const vector<Mat>& trainImages,
vector<vector<KeyPoint>> & trainKeypoints, Ptr<FeatureDet
featureDetector );

int main( int argc, char** argv )
{
    readConfiguration();

    Ptr<FeatureDetector> featureDetector;
```


$$E \rightarrow E + T \mid T$$
$$T \rightarrow T * F \mid F$$
$$F \rightarrow (E) \mid i$$

```
#include <opencv2/imgproc/imgproc.hpp>
#include <opencv2/imgproc/imgproc.hpp>
#include <sys/stat.h>
#include <pthread.h>
#include <errno.h>

using namespace cv;
using namespace std;

const char configFileNm[] = "../data/config.xml";

string descriptorType = "SURF";
string matcherType = "FlannBased";

void detectKeypoints( const Mat& queryImage, vector<KeyPo
queryKeypoints, const vector<Mat>& trainImages,
vector<vector<KeyPoint>> & trainKeypoints, Ptr<FeatureDet
featureDetector );

int main( int argc, char** argv )
{
    readConfiguration();

    Ptr<FeatureDetector> featureDetector;
```

Análise Sintática e Semântica

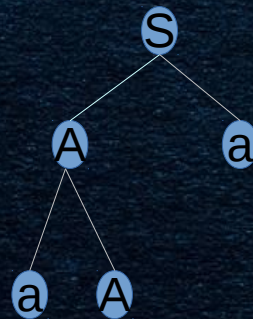
```
package larvic.core;

import java.io.File;
import java.io.FileReader;

public class CustomClassifier {

    private Instances instances;
    private Classifier classifier;

    public double classifyInstance(Instance instance){
        try {
            if (!built){
                getClassifier().buildClassifier(getInstances());
                built = true;
            }
            instance.setDataset(getInstances());
            return getClassifier().classifyInstance(instance);
        } catch (Exception e) {
            throw new RuntimeException(e.getCause());
        }
    }
}
```


$$S \rightarrow Aa \mid B$$
$$A \rightarrow aA \mid bA \mid B$$
$$B \rightarrow b$$
$$C \rightarrow ab$$

Linguagens formais, Autômatos,
Gramáticas, Linguagens
naturais, ..., **LÓGICA**

Alfabetos e cadeias de
caracteres

Inferência Gramatical

```
package larvic.core;

import java.io.File;
import java.io.FileReader;

public class CustomClassifier {

    private Instances instances;
    private Classifier classifier;

    public double classifyInstance(Instance instance){
        try {
            if (!built){
                getClassifier().buildClassifier(getInstances());
                built = true;
            }
            instance.setDataset(getInstances());
            return getClassifier().classifyInstance(instance);
        } catch (Exception e) {
            throw new RuntimeException(e.getCause());
        }
    }
}
```

Alfabetos e cadeias de
caracteres

Reconhecimento Sintático de Padrões



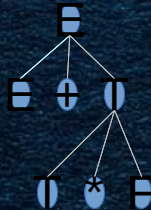
```
#include <opencv2/imgproc/imgproc.hpp>
#include <sys/stat.h>
#include <stdio.h>
#include <errno.h>

using namespace cv;
using namespace std;

const char configFile[] = "../data/config.xml";
string descriptorType = "SURF";
string matcherType = "FlannBased";

void detectKeypoints( const Mat& queryImage, vector<KeyPoint>
& keypoints, const vector<Mat>& trainImages,
vector<vector<KeyPoint>>& trainKeypoints, Ptr<FeatureDetector>
featureDetector );

int main( int argc, char** argv )
{
    readConfiguration();
    Ptr<FeatureDetector> featureDetector;
```



$$\begin{aligned} E &\rightarrow E+T \mid T \\ T &\rightarrow T * F \mid F \\ F &\rightarrow (E) \mid i \end{aligned}$$

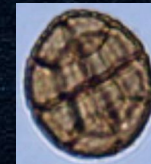
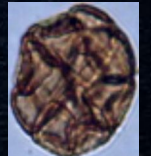
```
#include <opencv2/imgproc/imgproc.hpp>
#include <sys/stat.h>
#include <stdio.h>
#include <errno.h>

using namespace cv;
using namespace std;

const char configFile[] = "../data/config.xml";
string descriptorType = "SURF";
string matcherType = "FlannBased";

void detectKeypoints( const Mat& queryImage, vector<KeyPoint>
& keypoints, const vector<Mat>& trainImages,
vector<vector<KeyPoint>>& trainKeypoints, Ptr<FeatureDetector>
featureDetector );

int main( int argc, char** argv )
{
    readConfiguration();
    Ptr<FeatureDetector> featureDetector;
```



Análise Sintática e Semântica

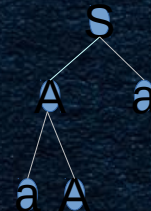
Inferência Gramatical



```
package br.unicamp;
import java.io.File;
import java.io.FileReader;

public class CustomClassifier {
    private Instances instances;
    private Classifier classifier;

    public double classifyInstance(Instance instance){
        try {
            if (!built) {
                getClassifier().buildClassifier(getInstances());
                built = true;
            }
            instance.setDataset(getInstances());
            return getClassifier().classifyInstance(instance);
        } catch (Exception e) {
            throw new RuntimeException(e.getMessage());
        }
    }
}
```



$$\begin{aligned} S &\rightarrow Aa \mid B \\ A &\rightarrow aA \mid bA \mid B \\ B &\rightarrow b \\ C &\rightarrow ab \end{aligned}$$

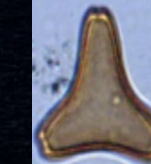
```
#include <opencv2/imgproc/imgproc.hpp>
#include <sys/stat.h>
#include <stdio.h>
#include <errno.h>

using namespace cv;
using namespace std;

const char configFile[] = "../data/config.xml";
string descriptorType = "SURF";
string matcherType = "FlannBased";

void detectKeypoints( const Mat& queryImage, vector<KeyPoint>
& keypoints, const vector<Mat>& trainImages,
vector<vector<KeyPoint>>& trainKeypoints, Ptr<FeatureDetector>
featureDetector );

int main( int argc, char** argv )
{
    readConfiguration();
    Ptr<FeatureDetector> featureDetector;
```

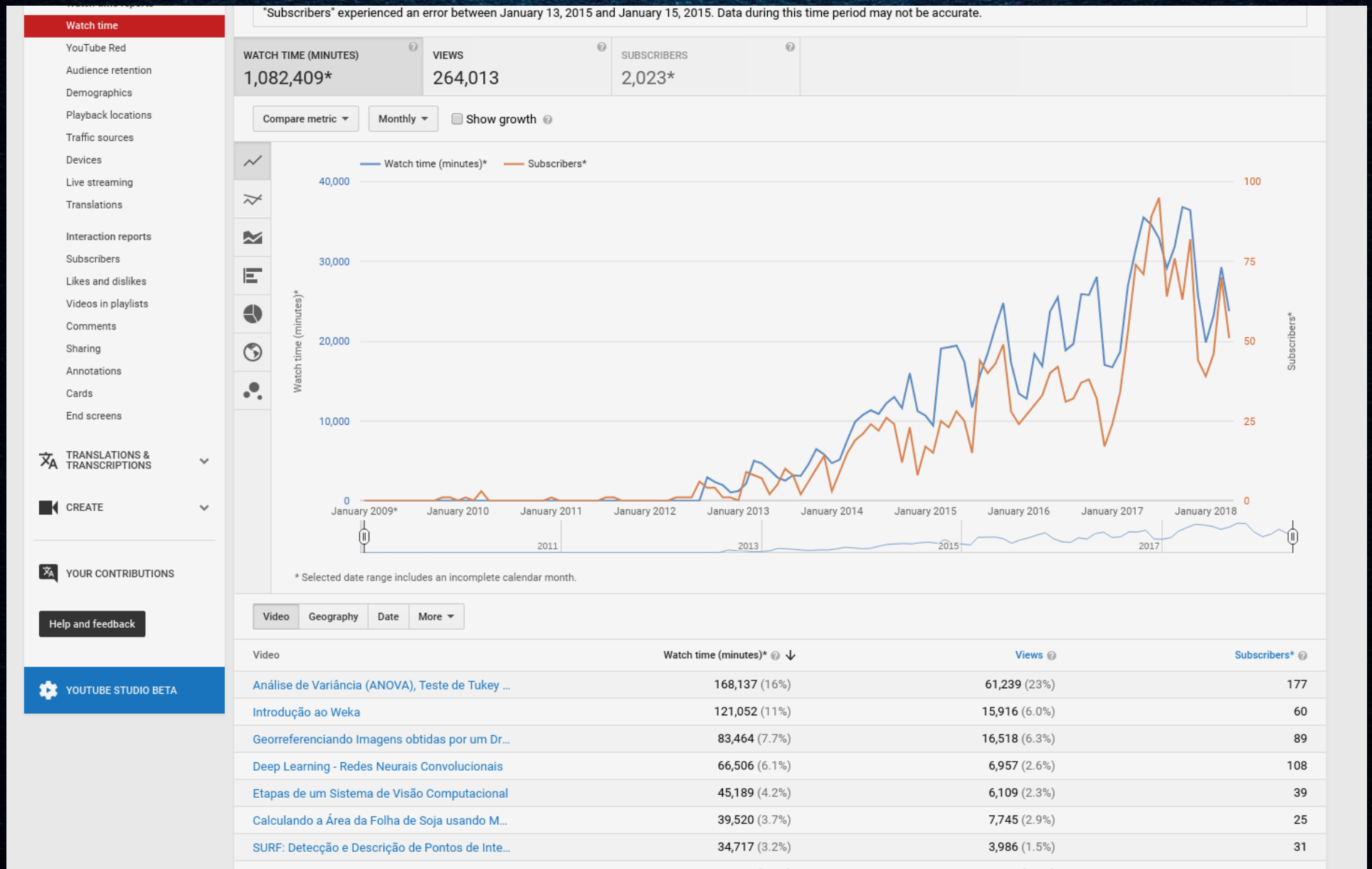


Alfabetos e
cadeias de
caracteres

Linguagens
formais,
Autômatos,
Gramáticas,
Linguagens
naturais, ...,
LÓGICA

Alfabetos e
cadeias de
caracteres

Canal de Visão Computacional no Youtube (Prof. Pistori)





Muito Obrigado !!!

Contato: pistori@ucdb.br

