

Relatório Final

**Andre J.S. Santos¹, Darlindo R.A. Júnior¹, Josinete R. Capim¹,
Lucas M.S. Carvalho¹, Lohann S.A. Paiva¹, Mercedes M.B Diniz¹**

¹Instituto de Tecnologia – Universidade Federal do Pará(UFPA)
Caixa Postal 479 – 66075-110 – Belém – PA – Brasil

andre.santos@itec.ufpa.br, darlindo.junior@itec.ufpa.br,
josinete.capim@itec.ufpa.br, lucas.souza.carvalho@itec.ufpa.br
lohann23@gmail.com, mercedes.diniz@itec.ufpa.br,

Abstract. *This report has evaluative finalists for a curricular component, Projects II, and corresponds to the final stage of the Robot 007.1 project. Here we will present the route of creating the system, encompassing its modeling, carrying out simulations, developing the circuit and software, analyzing the results and making adjustments. We will also present the instructions required for using the system through the user manual.*

Resumo. *Este relatório tem fins avaliativos para a componente curricular, Projetos II, e corresponde a etapa final do projeto Robô 007.1. Aqui apresentaremos o percurso da criação do sistema, englobando a modelagem do mesmo, realização das simulações, desenvolvimento do circuito e software, análise dos resultados e ajustes. Além disso apresentaremos as instruções necessárias para o utilização do sistema através do manual do usuário.*

1. Introdução

Como proposto inicialmente, a ideia do projeto é um robô disfarçado para segurança residencial que atenderia à dois modos de ação, um modo principal e um modo secundário. Dado as limitações impostas pela falta de alguns recursos, muito do sistema inicial foi modelado de forma simplificada, assim como a logística do cenário em que o sistema será utilizado. Vale destacar também, que não foi possível testar os códigos base do protótipo devido a falha no módulo GPRS GSM SIM800c. Para compensar tal fato foi criado um sistema abstrato para ilustrar a logica do sistema.

2. Metodologia e Organização do trabalho em equipe

FUNÇÃO DOS INTEGRANTES DA EQUIPE DE SISTEMA DE VIGILÂNCIA RESIDENCIAL	
MERCEDES MARIA BARBOSA DINIZ - 201906840030 FUNÇÃO GERAL: DESENVOLVEDORA PLENA	Desenvolveu a programação, mantendo o controle de versão e o repositório com o uso do Git e Git Hub. Desenvolveu a apresentação de desenvolvimento. Pesquisou o custo dos recursos e serviços do módulo principal. Apresentou o Desenvolvimento durante a aula. Fez a formatação de todos os relatórios e escreveu a Introdução, Padrão GSM, Comandos AT, Mensagens SMS e MMS, Módulo Principal, Sistema Básico de Detecção, Sistema de Comunicação, Circuito base do módulo principal, Software do Módulo Principal (Tópicos 4.1, 4.2, 4.3, 5, 5.1, 5.2, 5.3 e 5.4) do Relatório de Desenvolvimento. Escreveu a Introdução, Apresentação do produto e Conclusão do primeiro Relatório (Planejamento).
JOSINETE ROCHA CAPIM - 201906840040 FUNÇÃO GERAL: DESENVOLVEDORA PLENA	Desenvolveu a programação, mantendo o controle de versão e o repositório com o uso do Git e Git Hub. Desenvolveu a apresentação de desenvolvimento. Pesquisou o custo dos recursos e serviços do módulo principal. Apresentou o Desenvolvimento durante a aula. Escreveu sobre Referências Teóricas, Padrão GSM, Comandos AT, Mensagens SMS e MMS, Módulo Principal, Sistema Básico de Detecção, Sistema de Comunicação, Circuito base do módulo principal, Software do Módulo Principal, Abstração do Sistema, Idealização do Produto Final. (Tópicos 4.0, 4.1, 4.2, 4.3, 5, 5.1, 5.2, 5.3 e 5.4, 6.0 e 8.0) do Relatório de Desenvolvimento.
LUCAS MOZART DE SOUZA - 201906840034 FUNÇÃO GERAL: DESENVOLVEDOR PLENO	Desenvolveu a programação da detecção (Módulo 2), mantendo o controle de versão e o repositório com o uso do Git e Git Hub. Executou simulações do módulo 2. Escreveu no relatório de desenvolvimento no tópico 4.5 e tópico 7 sobre o Machine Learning.
ANDRE JORDAN SANTOS DOS SANTOS - 201906840038 FUNÇÃO GERAL: DESENVOLVEDOR PLENO	Ajudou a desenvolver a programação da detecção (Módulo 2). Executou simulações do módulo 2. Apresentou o módulo 2 no desenvolvimento durante a aula. Escreveu no relatório de desenvolvimento no tópico 4.5 e tópico 7 sobre o Machine Learning.
DARLINDO RIBEIRO DE ALMEIDA JUNIOR - 201907040048 FUNÇÃO GERAL: GERENTE DO PROJETO E AUXILIAR TÉCNICO	Produziu o Slide da primeira apresentação. Montagem Física do circuito da Comunicação (GSM) junto a Detecção (PIR). Pesquisou o custo dos recursos e serviços do módulo principal. Escreveu no relatório sobre a Metodologia e Organização do trabalho em equipe e o tópico 4.4- Tecnologia Infravermelha. Escreveu sobre a Introdução, Materiais e Cronograma Geral no Relatório de Planejamento.
LOHANN SILVA AMARAL PAIVA - 201907040041 FUNÇÃO GERAL: DESENVOLVEDOR PLENO	Pesquisou o custo dos recursos e serviços do módulo principal. Escreveu no relatório referente ao tópico 4.4 sobre Tecnologia Infravermelha. Ajudou na construção do slide da primeira apresentação.

Figura 1. Descrição das funções e Atividades dos Integrantes da Equipe

3. Referencias Teóricas

Nesta seção, são apresentados os conceitos importantes para o desenvolvimento do sistema, tanto em termos de tecnologias, como em termos de protocolos.

3.1. Padrão GSM

Criada em 1982, a rede GSM foi criada para acabar com a incompatibilidade nas redes dos sistemas de telefonia móvel. Um grupo de estudos Europeu foi criado para desenvolver um sistema móvel padrão o Group Special Mobile (GSM). Considerada tecnologia de

segunda geração, a mesma é usada para transmissão de voz e serviços de dados móveis. Suporta chamadas de voz e dados como o envio de SMS, com velocidade de transferência de até 9,6 Kbps.[Silva]

3.2. Comandos AT

São também conhecidos como conjunto de comandos Hayes, formando um conjunto de comandos específicos para modems. Os comandos AT consistem de uma série de curtas palavras de texto que se combina para produzir um conjunto completo de comandos para operações como a discagem, desligamento, e alteração dos parâmetros da conexão.[Salvador et al. 2008]

3.3. Mensagens SMS e MMS

As mensagens de texto (Short Message Service) estão entre as formas baratas e rápidas de se comunicar. O serviço de mensagens curtas SMS é um protocolo de comunicação que permite a troca de mensagens curtas de texto entre telefones celulares.

Para o funcionamento desse serviço, as mensagens são trocadas entre telefones celulares num esquema “store-and-forward”, ou seja, um aparelho envia uma mensagem para outro, sendo que esta mensagem será recebida por um centro de controle da operadora do serviço antes de ser transmitida ao seu destinatário final. O centro de controle citado é chamado de SMSC (Short Message Service Center) e é responsável por todas as tarefas relativas ao processamento das mensagens, como roteamento, cobrança e entrega das mensagens.[Christiano Freitas de Souza]

O SMS sofreu uma evolução para o MMS (Multimedia Message System) ou Sistema de Mensagens Multimídia, o mesmo inclui quatro tipos básicos de serviços: mensagens multimídia originadas no celular; mensagens multimídia terminadas no celular; mensagens multimídia originadas em uma aplicação; mensagens multimídia terminadas em uma aplicação. A central “store and forward” desse serviço é o MMS Center. Quando o mesmo recebe a mensagem ela a armazena em seu banco de dados; o MMS Center envia a mensagem quando a estação móvel pede; se a mensagem não pu-

der ser entregue com sucesso, ela é armazenada para ser enviada posteriormente; depois de uma entrega bem sucedida, a mensagem é apagada do banco de dados do MMS Center.[Christiano Freitas de Souza]

3.4. Tecnologia Infravermelha

Todos os objetos que possuam temperatura acima do zero absoluto emitem energia térmica na forma de radiação. Essa radiação, no geral, é invisível ao olho humano, pois é emitida em comprimentos de onda na faixa do infravermelho, a mesma porém pode ser detectada por sensores especiais, como o PIR.

O “P” da sigla PIR significa passive (passivo), indicando que o dispositivo não gera nem irradia energia IR em seu processo de detecção, operando exclusivamente por meio da detecção de energia infravermelha emitida pelos objetos.

Os comprimentos de onda infravermelhos emitidos através do sensor PIR são feitos por meio do aquecimento do quartzo que está dentro dele, e variam entre 3,5 μm e 0,9 μm . Dependendo do tipo de material a ser aquecido é possível utilizar diferentes tipos de comprimentos de onda de infravermelhos para alcançar a máxima propagação de energia e obter um processo de aquecimento mais rápido e eficaz.

3.5. Machine Learning

Antes de falarmos sobre Machine Learning é relevante explicar o que é inteligência artificial, que seria, uma de suas inúmeras descrições, um jeito de fazer com que o computador “pense” como uma pessoa para executar tarefas, Machine Learning seria um subconjunto desse grande conjunto que é a inteligência artificial. Machine Learning é também conhecido como aprendizado de máquina ou aprendizado automático é um sistema que pode modificar seu comportamento de forma autônoma de acordo com os dados que ele teve contato, sua experiência, com o mínimo de interferência humana possível ou nenhuma.[Alecrim]

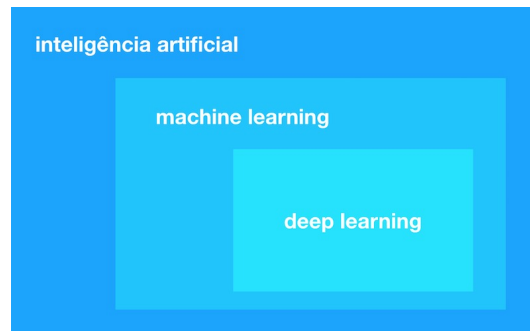


Figura 2. Subconjuntos da Inteligência Artificial

O Machine learning é usado em uma variedade de tarefas computacionais onde criar e programar algoritmos explícitos é impraticável. Com as principais aplicações em filtragem de spam, reconhecimento ótico de caracteres, diagnósticos médicos, bioinformática, reconhecimento de fala, visão computacional e locomoção de robô. Com a evolução da tecnologia o machine learning gerou alguns ramos como deep learning ou aprendizagem profunda, que faz parte do próprio machine learning, é uma área que se inspira na rede neural do cérebro humano e desenvolve o conceito das redes neurais artificiais com múltiplas camadas, que são os “neurônios” dela. Isso potencializa o machine learning normal dando uma quantidade significativamente maior de processamento o que permite ao algoritmo trabalhar com situações mais complexas como reconhecimento de fala ou imagens.[SALESFORCE]

4. Módulo Principal

Este modo de operação tem como objetivo fornecer as funcionalidades mínimas que caracterizam o sistema proposto. Neste cenário, o usuário iria acionar esse módulo (que estaria posicionado de forma estática em um local adequado), antes de sair do ambiente em questão. A lógica aqui é simples, se o sensor de presença (PIR) detectar algo, ele mandará uma mensagem de alerta por SMS e uma foto. O usuário de posse dessas informações decidirá se chama o serviço de segurança ou apenas ignora o ocorrido.

4.1. Sistema de Comunicação

Para cumprir com essa funcionalidade foi usado o módulo GPRS GSM SIM800c para o Arduino, o mesmo é utilizado para comunicação via dados GSM/GPRS e necessita de um chip de operadora de telefonia móvel para comunicação. Como apresentado anteriormente, as redes GSM são uma tecnologia digital usada para transmissão móvel de voz e dados.

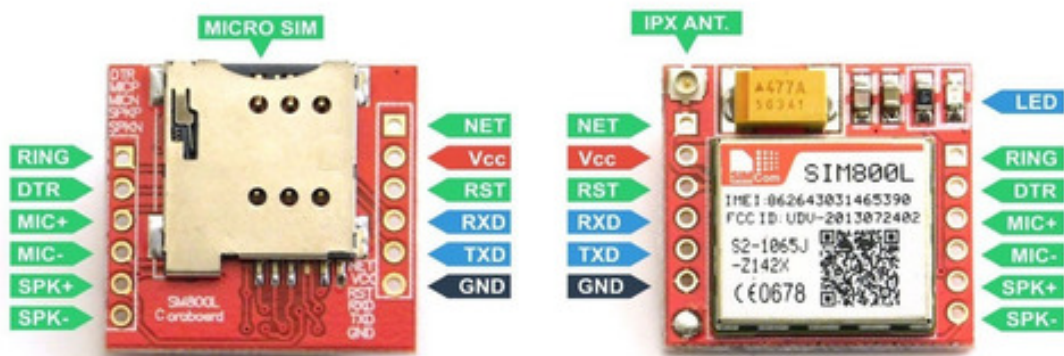


Figura 3. Módulo GPRS GSM SIM800

4.2. Sistema Básico de Detecção

Em nosso projeto, vamos utilizar o sensor de movimento PIR HC-SR501, que pode ser encontrado facilmente, o mesmo é baseado na tecnologia infravermelha, sendo um módulo de controle automático, com alta sensibilidade e confiabilidade, que opera em voltagem ultrabaixa. O mesmo é amplamente utilizado em vários equipamentos elétricos de detecção automática. O sensor tem três conexões, sendo elas: Um GND (terra), um de voltagem (5V-20V) e um OUTPUT.[de Silício]

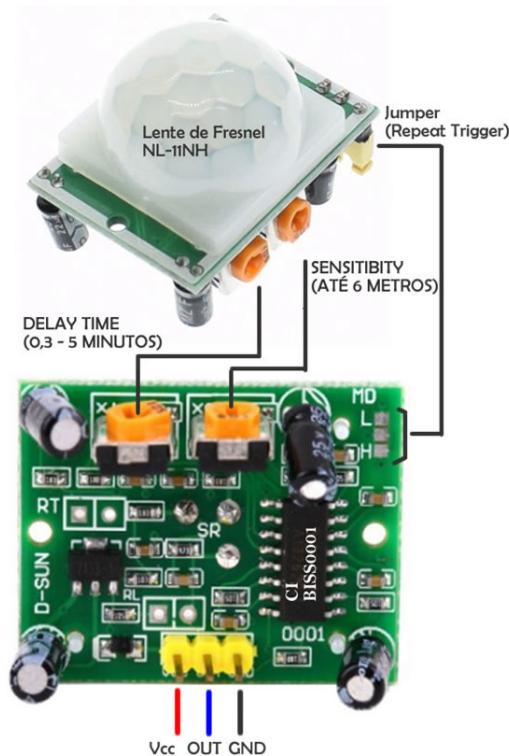


Figura 4. Sensor PIR

Fisicamente, o sensor PIR é composto de duas partes construídas com material sensível à radiação infravermelha. Cada metade consegue detectar a radiação IR até uma distância específica, que basicamente dita a sensibilidade do sensor. Quando não há movimento na área de visão do sensor, as duas metades detectam a mesma quantidade de radiação presente no ambiente. Porém, quando um objeto quente (como uma pessoa ou animal) passa na frente do sensor, a radiação emitida por esse corpo é detectada primeiramente por uma das metades do PIR, causando o que chamamos de mudança diferencial positiva entre as duas metades. Quando o corpo aquecido sai da área de detecção, ocorre o inverso – mudança diferencial negativa.[dos Reis]

Quando a quantidade de radiação infravermelha muda repentinamente, o sensor é acionado, o que ocorre quando um objeto se move em frente ao campo de visão do sensor, causando um transiente no sinal. Quanto mais quente o objeto, mais radiação ele emite, e mais facilmente é detectado.

4.3. Circuito Base do Módulo Principal

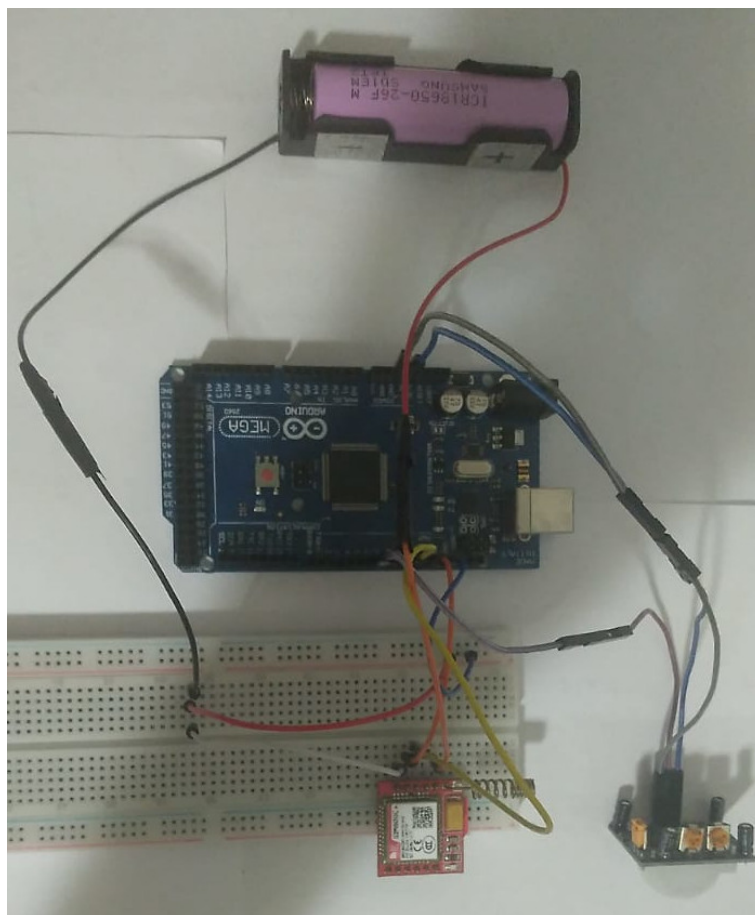
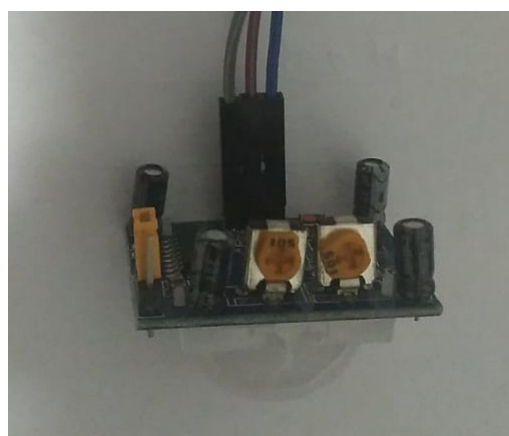


Figura 5. Primeiro protótipo desenvolvido



(a) Módulo SIM800c



(b) Sensor PIR

Figura 6. Componentes Principais do Protótipo

PINAGEM DAS PORTAS DO PROJETO				
GND ARDUINO	JUMPER MACHO AZUL ESCURO	POSITIVO DA PROTOBOARD	J.VERM - FILA H	J.BRANCO GND GSM
PORTA SERIAL 11 ARDUINO	JUMPER MACHO LARANJA	TX DO GSM SIM800C		
PORTA SERIAL 10 ARDUINO	JUMPER MACHO AMARELO	RX DO GSM SIM800C		
POSITIVO DA BATERIA	JUMPER MACHO LARANJA 2	VCC DO GSM SIM800C		
NEGATIVO DA BATERIA	JUMPER MACHO PRETO FILA J	MESMA COLUMNA DO VERM e BRANCO		
GND POWER ARDUINO	JUMPER MACHO-FÊMEA CINZA	GND DO PIR		
5V DO ARDUINO	JUMPER MACHO-FÊMEA AZUL 2	VCC DO PIR		
PORTA SERIAL 7 ARDUINO	JUMPER ROXO MACHO-FÊMEA	SAÍDA DE SINAL DO PIR		

Figura 7. Pinagem e Conexões

4.4. Software do Módulo Principal

Com base na plataforma de prototipagem eletrônica de hardware, Arduino, e na sua IDE, todo o código desenvolvido para o módulo principal foi baseado na linguagem de programação C++. Os códigos referentes ao protótipo do módulo principal são o "prototipo01.ino" e o "Teste01-GSM-MMS1.ino", os mesmos foram separados para fins de organização. Além disso, as principais funções, que são encarregadas da comunicação do sistema com o usuário, contam com o uso dos comandos AT para lidar com o módulo GPRS GSM SIM800c. A seguir será apresentado uma lista com os comandos AT, presentes nos códigos citados.

A logica de programação do código "prototipo01.ino" é bem direta, se o sensor de presença (PIR) detectar algo ele ficará em nível "High", e essa informação é o parâmetro de um "if". Dentro dessa estrutura de decisão temos instruções para que o dispositivo envie um SMS de alerta para o usuário (o número do mesmo deve ser cadastrado com antecedência). Nesse ponto, o sistema espera uma resposta por SMS, do usuário, para atender sua decisão. O mesmo pode escolher entre ignorar ou acionar a função de escuta, que aqui seria apenas o dispositivo ligando para o usuário e usando um microfone (instalado no módulo GSM) para o mesmo ouvir o ambiente. Em caso de algo suspeito o próprio morador deverá ligar para a polícia.

COMANDOS	DESCRIÇÃO
AT+CMGF=1	Modo Texto
AT+CNMI=2,2,0,0,0	Mostra Mensagens
ATX4	Habilita a detecção do ocupado e de linha e informa a velocidade de conexão
AT+COLP=1	Ativa a apresentações do COL (linha conectada)
AT+CMGS="+5511numero"	Envio de Mensagem
ATH0	Desligar (colocar no gancho)
ATD+telefone	Discagem

Figura 8. Comandos AT utilizados

Quanto ao código "Teste01-GSM-MMS1", desenvolvido exclusivamente para exemplificar a viabilidade da função de enviar uma imagem por MMS. A primeira etapa é verificar se o dispositivo móvel oferece suporte GPRS/EDGE. Somente neste caso, poderá ocorrer o envio e recebimento de mensagens mms. A primeira função serve como parâmetro para inicializar as configurações, assim estabelecer uma conexão com o MMS Center. Após a confirmação, a mensagem será enviada ao destinatário, por seguinte, ao receber a resposta, sairá do modo mms.

COMANDOS	DESCRIÇÃO
AT + CMMSINIT	Inicializar função MMS
AT + CMMSCURL = " https://ibb.co/6wSrbMj "	Defina o URL do centro de MMS
AT + CMMSCID = 1	Parâmetros de rede para MMS
AT + CMMSPROTO = " 10.0.0.172 ", 80	Parâmetro de protocolo e proxy MMS
AT + CMMSENDCFG = 6,3,0,0,2,4,1,0	Parâmetros para enviar MMS
AT + SAPBR = 3,1, " CONTYPE ", " GPRS "	Definir parâmetro portador
AT + SAPBR = 3,1, " APN ", " CMWAP "	Definir o contexto do portador
AT + SAPBR = 1,1	Contexto do portador ativo
AT + SAPBR = 2,1	Ler parâmetro do portador
AT + CMMSEDIT = 1	Entrar ou sair do modo de edição
AT + CMMSDOWN = " PIC ", 12963,20000, "	Baixe os dados do arquivo/título do UART
AT + CMMSRECP = "13918181818 "	Adicionar destinatários
AT + CMMSVIEW	O MMS no buffer e mostrando as informações
AT + CMMSSEND	Começar a enviar MMS
AT + CMMSEDIT = 0	Entrar ou sair do modo de edição
AT + CMMSTERM	Sair da função MMS

Figura 9. Comandos AT utilizados

5. Abstração do Sistema

Para simular de uma possível invasão residencial, o Sensor de presença Pir é essencial no projeto para detectar algum movimento. O modulo pode receber ajustes relacionado a sensibilidade e o tempo que o sinal enviará ao arduino, quando recebido o sinal 1(HIGH) o Buzzer ligará numa frequência de 1500 Hetz. Assim, o proprietário da casa recebe a mensagem de um possível arrombamento em seu domicílio. Na mensagem tem a imagem e duas opções: Apertando o botão 1, o Led Verde acende, informando o descarte do alerta; Apertando o botão 2, o Led amarelo acende, ligando para o proprietário. Assim, o mesmo poderá ouvir o ocorrido.

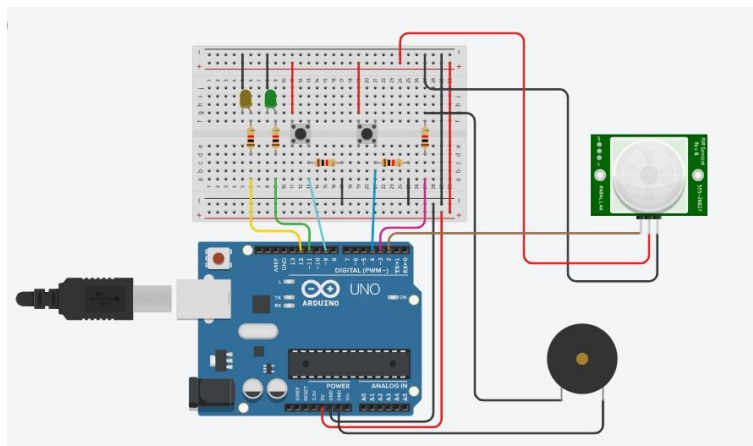


Figura 10. Circuito ilustrativo

6. Módulo Secundário

Este modo de operação tem como objetivo fornecer as funcionalidades mais avançadas, que são o diferencial do sistema.

Para essa parte do projeto que compreende a identificação de armas através de uma câmera foi usada a linguagem de programação python (com a IDE Pycharm) , uma de suas bibliotecas chamada OpenCV(Open Source Computer Vision) e um algoritmo da mesma chamado Haar Cascade para detectar os objetos que queremos.

O objetivo inicial era usa uma câmera do próprio robô para enviar esse vídeo a um computador para que o mesmo rodasse os códigos em python e fizesse todo o processamento que o arduino não seria capaz de fazer e então enviasse de volta para o robô o resultado caso encontrasse o objeto alvo. No entanto, optamos por simplificar o projeto devido ao tempo e trabalha com essa parte de machine learning como está descrito abaixo.

6.1. Simulação

Para a simulação usaremos uma webcam em um computador para simular a câmera do robô e a troca de informações entre o robô e o computador. Para começar, nós temos que treinar o nosso próprio haar cascade para ele detectar o objeto que queremos, armas, e não outros (cômodos e pessoas), para isso passaremos para o computador inúmeras imagens das duas situações anteriores totalizando mais ou menos 18 mil imagens de 7

datasets (conjunto de dados, retirados do site Kaggle)[Vasko]. Com essas imagens em mãos dentro do projeto do Pycharm e com as bibliotecas do openCV devidamente instaladas criaremos dois códigos que estão abaixo :

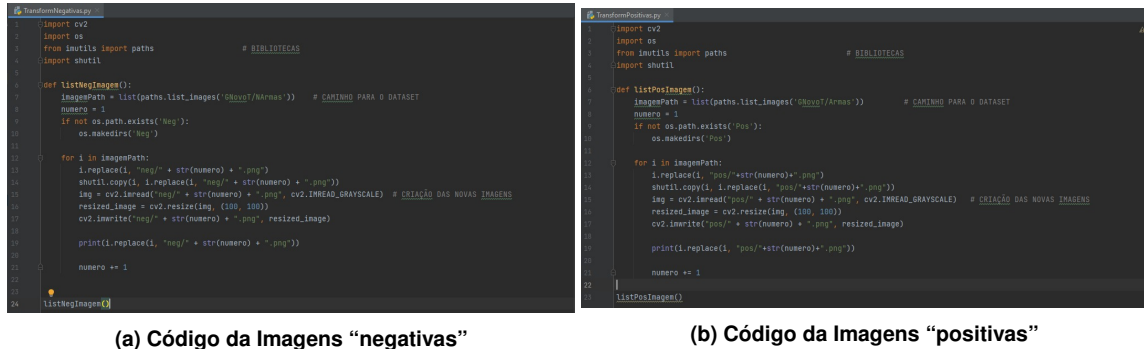


Figura 11. Códigos Base

Os códigos acima servem para criar copias das imagens do nosso dataset , um deles criará imagens “negativas”, o que não queremos detectar (para isso usamos fotos de cômodos de casas e pessoas), e o outro criará imagens “positivas”, o nosso objeto alvo (pistolas), essas imagens serão passadas para escala de cinza e redimensionadas (o openCV trabalha melhor com esses tipos de imagem). Feito isso o próximo passo é rodar o código à esquerda:

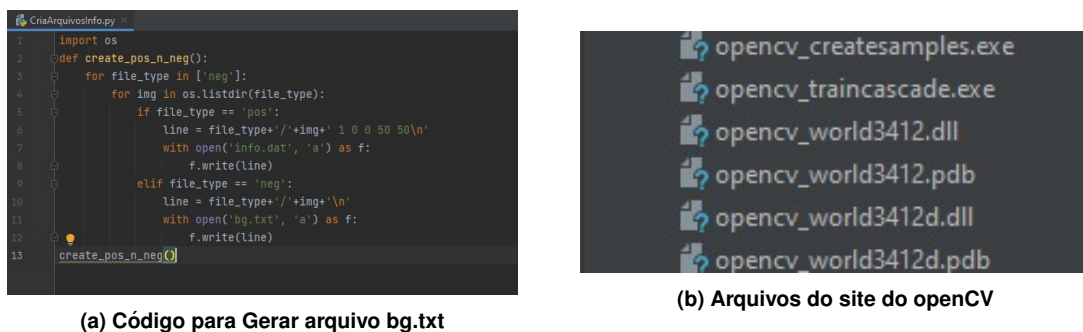


Figura 12. Passos Fundamentais

Isso vai gerar um arquivo bg.txt que contém o nome de todas as fotos “negativas”, esse arquivo será necessário para o nosso haar cascade. Para o próximo passo usaremos um simulador de linux chamado Cmder (isso vai deixa o processo mais fácil) e também baixaremos os arquivos ilustrados na imagem ”b”acima, diretamente do site do openCV

e colocaremos eles na pasta do nosso projeto do pycharm.

Agora , inicializaremos o cmd e usaremos o comando “cd” para definir como diretório a pasta do projeto do pycharm, feito isso usaremos três comandos :

```
opencv_createsamples.exe -img BASE.jpg -bg bg.txt -info info/info.lst -maxxangle 0.5 -maxyangle 0.5 -maxzangle 0.5 -num 8374  
opencv_createsamples.exe -info info/info.lst -num 8374 -w 20 -h 20 -vec positives.vec  
opencv_traincascade.exe -data data -vec positives.vec -bg bg.txt -numPos 8374 -numNeg 4187 -numStages 10 -w 20 -h 20
```

Figura 13. Comandos

O primeiro comando usara uma imagem que chamamos de BASE e deve ser colocado também na pasta geral do projeto, que no nosso caso será :



Figura 14. Arma

Quando rodamos esse comando, ele vai gera uma pasta info com um arquivo info.txt e mais ou menos 90 por cento (8374) do número de imagens “negativas” em novas imagens, sendo elas as mesmas negativas no entanto com a imagem “BASE” por cima em ângulos diferente, como os exemplos :



(a) Cenário 1



(b) Cenário 2

Figura 15. Imagem Base em Ângulos Distintos

Essas novas imagens também serão necessárias para treinar o modelo. O segundo comando servirá para criar um vetor, um arquivo .vec , que também será necessário para o haar cascade. E, por fim, nós treinaremos o modelo com o último comando, usaremos o valor de 90 por cento das imagens negativas de antes para as imagens positivas (-numPos) e para as imagens negativas desse comando(-numNeg) usaremos metade das -numPos.[Larxel] Feito isso o nosso haar cascade está pronto, se parecerá com a imagem abaixo e devera está guardado em uma pasta chamada data que nós criamos previamente na pasta do projeto do pycharm.

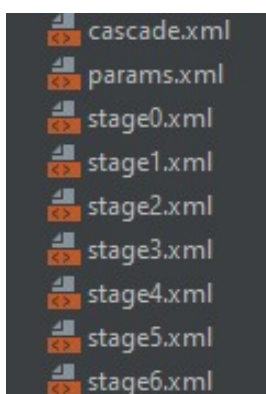
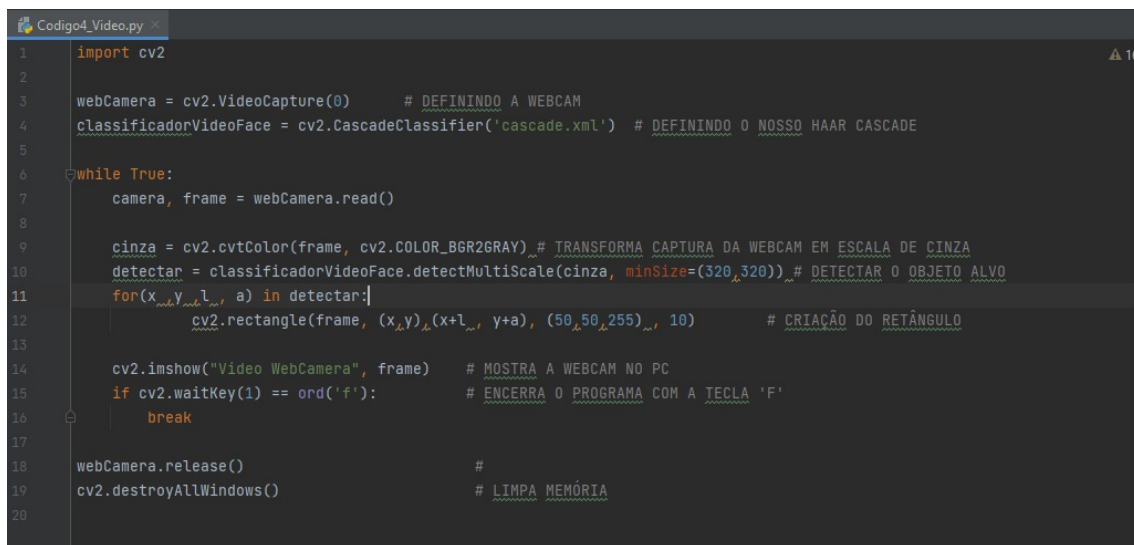


Figura 16. Haar Cascade

Durante o treinamento houveram alguns erros que nós resolvemos reiniciando o comando variando um pouco os números do numPos e numNeg , no caso ele reinicia

o treinamento de acordo com o estágio que parou, o que facilitou bastante. Feito tudo isso agora basta criar um código em python com as funções da biblioteca do openCV que acesse a webcam, faça uso do nosso haar cascade e desenhe um retângulo no nosso objeto alvo, que é o código a abaixo :

A screenshot of a code editor window titled 'Codigo4_Video.py'. The code is a Python script that uses OpenCV to access a webcam and detect faces using a Haar Cascade classifier. The script includes comments in Portuguese explaining each step. The code is as follows:

```
1 import cv2
2
3 webCamera = cv2.VideoCapture(0) # DEFININDO A WEBCAM
4 classificadorVideoFace = cv2.CascadeClassifier('cascade.xml') # DEFININDO O NOSSO HAAR CASCADE
5
6 while True:
7     camera, frame = webCamera.read()
8
9     cinza = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY) # TRANSFORMA CAPTURA DA WEBCAM EM ESCALA DE CINZA
10    detectar = classificadorVideoFace.detectMultiScale(cinza, minSize=(320,320)) # DETECTAR O OBJETO ALVO
11    for(x,y,l,a) in detectar:
12        cv2.rectangle(frame, (x,y),(x+l,y+a), (50,50,255), 10) # CRIAÇÃO DO RETÂNGULO
13
14    cv2.imshow("Video WebCamera", frame) # MOSTRA A WEBCAM NO PC
15    if cv2.waitKey(1) == ord('f'): # ENCERRA O PROGRAMA COM A TECLA 'F'
16        break
17
18 webCamera.release() #
19 cv2.destroyAllWindows() # LIMPA MEMÓRIA
20
```

Figura 17. Código com as funções do openCV e acesso a Webcam

O nosso modelo , infelizmente , não é perfeito , ele detecta muitas coisas que não devia e as vezes também não detectar o que deveria, em grande parte isso deve a dois fatores , o primeiro foi a quantidade muito pequena de imagens usadas para o projeto devido à dificuldade de achar muitas imagens desse tipo e ao fato de que levaria muito tempo com o computador no máximo da sua capacidade para processar grandes quantidades de fotos e o segundo fator foi a falta de boas imagens (com somente o objeto alvo na foto ou quase isso) devido a situação de que a maioria dos dataset que achamos as fotos das armas sempre estavam acompanhadas de pessoas ou outras coisas ,o que causa interferência no treinamento e ao fato de que para realiza uma filtragem de forma eficiente das imagens também levaria muito tempo.

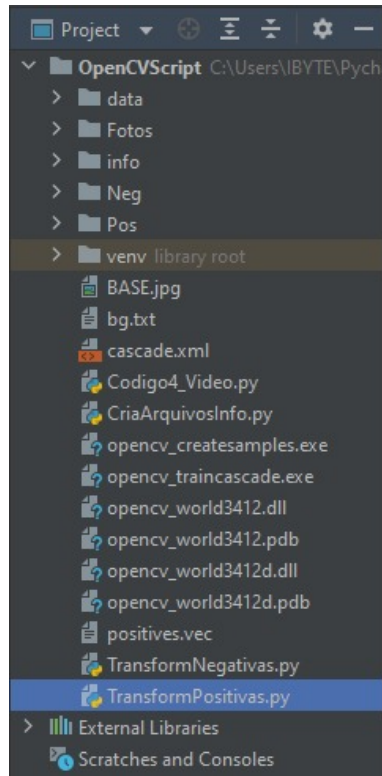


Figura 18. Pasta do Projeto



Figura 19. Simulação com uma Arma de Airsoft

7. Idealização do Produto Final

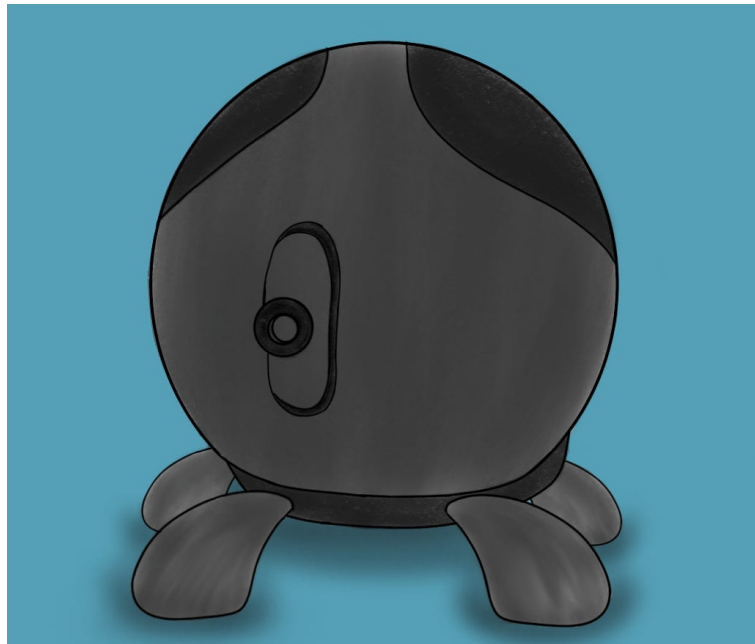


Figura 20. Design do Robô

8. Manual do Usuário

O manual aqui desenvolvido contará com a ideia do produto "finalizado", onde as funcionalidade do mesmo estariam integradas. Temos como objetivo exemplificar a instalação e as configurações iniciais necessárias para que o proprietário do Robô 007.1 possa utilizado em toda a sua capacidade.

8.1. Apresentação do Produto

O Robô 007.1 é um sistema de monitoramento para segurança residencial que opera em dois módulos, adaptados para cenários distintos. O mesmo pode ser usado para supervisionar um ambiente fechado, que deve está vazio e não pode ser invadido. Ou em uma situação mais complexa, onde ele fiscalizaria um ambiente movimentado em busca de possíveis armas de fogo.

O sistema finalizado contará, de forma básica, com uma câmera de qualidade, um módulo bluetooth como o HC-05, um sensor de presença (PIR) e uma módulo SIM800I

com um chip ativo e microfone e antena instalados, além de fontes de alimentação adequadas para manter o sistema com alguma autonomia.

8.2. Instalação e Configurações Iniciais

A localidade e posicionamento do equipamento faz muita diferença, o robô não foi projetado para se movimentar nessa versão simplificada, logo esse cuidado é de responsabilidade do usuário. Recomenda-se que o mesmo posicione o robô em um local em que ele não seja facilmente detectado, como em uma estante de altura mediana ou alta, em uma lateral. Os sensores e câmeras devem estar virado para entra/saída ou o que for de maior conveniência.

Para que a comunicação do robô com o usuário seja estabelecida é necessário o número do celular do mesmo seja cadastrado. Também é necessário conectar o arduino por bluetooth com um computador pessoal do próprio usuário, que possua os arquivos necessários para realizar o machine learning e envia a resposta para o arduino.

8.3. Acionando os Modos/Módulos de Operação

Os dois módulos de operação do sistema, descritos em tópicos anteriores, são acionados manualmente pelo proprietário, que deve seleciona-lo de acordo com o objetivo. O módulo primário é mais adequado para supervisionar um ambiente fechado que não deve ser violado. Já o módulo secundário é adaptado para trabalhar em um local dinâmico, fiscalizando possíveis indivíduos armados.

9. Conclusão

Em função do projeto desenvolvido, que teve como finalidade construir um protótipo miniaturizado de um projeto como robô vigilante, fácil interatividade e gerenciado por dispositivos móveis como smartphones e tablets, foi possível verificar que o objetivo principal do mesmo foi atingido.

10. Citações

[Analytics 2020a][Analytics 2020b][Analytics][Ahmadzada][aR5P1eDy][Criollo][Brigade]
[Dia a][Dia b][SAS][Howard][Larxel][opencv][Silva][desconhecido][ArduinoPortugal.pt]
[helios quartz]

Referências

Ahmadzada, A. People image dataset.

Alecrim, E. Machine learning: o que é e por que é tão importante.

Analytics, C. Python e opencv - instalando ferramentas treino haar cascade para analise de imagens.

Analytics, C. (2020a). Como analisar imagens usando python e opencv - pycharm parte 1-7.

Analytics, C. (2020b). Detecção de objetos com python opencv e yolo.

aR5P1eDy. Weapons datasets.

ArduinoPortugal.pt. Como ligar o sensor de movimento pir hc-sr501 com arduino.

Brigade, D. S. A diferença entre inteligência artificial, machine learning e deep learning.

Christiano Freitas de Souza, R. O. R. Mensagens multimídia – do sms ao mms.

Criollo, B. Dataset handguns 1seg.

de Silicio, V. Sensor de presença pir - hc-sr501.

desconhecido. Hc-sr501 passive infrared (pir) motion sensor.

Dia, C. T. Deep learning explicado.

Dia, C. T. A revolução das inteligências artificiais.

dos Reis, F. Como funciona um sensor de movimento pir – passive infrared.

helios quartz. Tecnologia de infravermelho.

Howard, J. Lsun bedroom scene 20

Larxel. Handgun detection.

opencv. Opencv modules.

SALESFORCE. Deep learning e machine learning: conheça as diferenças.

Salvador, C. J., de Medeiros, D. d. S., and da Silva, F. E. (2008). Transmissão de seqüências de imagens através da rede celular gsm/edge. *II Jornada da Produção Científica da Educação Profissional e Tecnológica da Região Sul, Pelotas*.

SAS. Machine learning. o que é e qual sua importância?

Silva, A. V. d. O. Alarme com ativação por sensor presencial e alerta via sms.

Vasko, S. Emulador de console portátil para windows.