

Universidade de São Paulo

Escola de Artes Ciências e Humanidades



**EACH**

---

**Desenvolvimento de Componentes de Visão  
Computacional para Programação por Blocos**

---

Docente responsável:  
Prof. Dr. Flavio Luis Coutinho.

## **Resumo**

Visando lidar com os desafios no aprendizado de programação para aplicativos voltada para a visão computacional, campo pertencente a área de Inteligência Artificial, este trabalho tem como objetivo a criação de extensões para o ambiente de programação em blocos App Inventor, muito utilizado por facilitar o desenvolvimento de aplicativos para iniciantes interessados na área de computação e crianças. Estas extensões pretendem permitir que conhecimentos acerca de algoritmos mais complexos no âmbito de IA sejam menos exigidos.

## **Justificativa**

Visão computacional está se tornando uma área cada vez mais utilizada na sociedade atual. Aplica-se a reconhecimento de sorriso em câmeras fotográficas, reconhecimento facial em sistemas de vigilância, reconhecimento de culturas de plantas em fotografias aéreas e outras.

Trabalhar nessa área requer conhecimento em computação, Inteligência Artificial, processamento de imagens e na área de aplicação. A conjunção das habilidades para o trabalho geralmente é conseguida formando equipes multidisciplinares e, mais recentemente, dando formação interdisciplinar aos profissionais.

Uma alternativa que poderia ser mais explorada é o uso de ferramentas visuais, como programação por blocos, para reduzir a exigência nas habilidades ligadas à tecnologia da informação e comunicação, consequentemente permitindo

que profissionais da área de aplicação possam usar ferramentas de visão computacional.

Programação por blocos é uma forma de criar programas que já mostrou que facilita/acelera o desenvolvimento de aplicativos. É fácil de usar, razoavelmente intuitivo, pode inclusive ser usado para ensinar crianças a programar - talvez o uso mais popular da programação por blocos. Esta forma de programar tem suas limitações: à medida que a quantidade de blocos aumenta a ferramenta perde a comodidade pela poluição visual e quantidade e amplitude de movimentos físicos (arrastar objetos na tela) necessários para construir os programas. Ainda assim, com as escolhas adequadas, pode ser uma ferramenta útil para o desenvolvimento de aplicações.

App Inventor (POKRESS, VEIGA, 2013) é um dos mais conhecidos ambientes de programação em blocos. Desenvolvido pelo MIT, busca facilitar o desenvolvimento de aplicativos que têm como base a linguagem Java. Esta plataforma faz uso do Blockly, uma biblioteca que adiciona um código representado por blocos que formam expressões lógicas quando anexados uns aos outros.

OpenCV (Bradski, 2000) significa *Open Source Computer Vision Library*, é uma biblioteca de software aberto para visão computacional e aprendizado de máquina (openCV, 2019). Originalmente desenvolvida na linguagem C/C++, por ser Open Source e ser constantemente otimizada, fez com que outras formas de aplicações com diferentes linguagens pudessem utilizá-la, tendo inclusive suporte ao Android. Dessa forma, possibilitou diversos avanços na pesquisa em visão computacional. Embora esteja disponível para ser incorporada às aplicações feitas no Android Studio, IDE oficial e profissional de aplicações Android, a biblioteca

OpenCV não oferece suporte para o App Inventor e nem para o appybuilder. Há projetos que tentaram implementar funcionalidades semelhantes, mas de forma limitada e desatualizada.

## Objetivos

1. Criar uma ou mais extensões de alguma ferramenta de programação por blocos que permitam a pessoas pouco experientes em linguagens de programação utilizar técnicas de Visão Computacional.
2. Divulgar e documentar o desenvolvimento de extensões e os dados coletados com o intuito de atrair desenvolvedores e interessados da área de computação e de áreas de aplicação.
3. Introduzir o aluno de graduação da USP em uma linha de pesquisa nas Áreas de Visão Computacional, Inteligência Artificial e Construção de Aplicativos.

## Materiais e métodos

Trata-se do desenvolvimento de componentes de software utilizando metodologia ágil. Nas primeiras semanas o estudante conhecerá colegas em diferentes estágios do seu trabalho de Iniciação Científica, que podem ser fontes de informação ou facilitadores para seu projeto. Nas semanas subsequentes, o estudante desenvolverá suas atividades recebendo metas semanalmente à medida que cumpre as metas anteriores, em um ciclo de desenvolvimento. Ao final espera-se entregar todos os resultados.

## **Detalhamento das atividades**

1. Integrar-se ao grupo de estudantes de Iniciação Científica e Colaboradores do orientador deste projeto:  
  
Participar de (poucas) reuniões gerais, conviver (muito) com os seus colegas no desenvolvimento de seu projeto usando a estrutura do laboratório, sempre que possível. Criar o hábito do encontro e trabalho no laboratório, coletivamente, com todos os estudantes e orientador.
2. Familiarizar-se com as ferramentas de desenvolvimento (Ferramenta de programação por blocos escolhida, OpenCV) :  
  
Ao longo do projeto aprender continuamente como utilizar as ferramentas para atingir metas colocadas semanalmente;
3. Construir um protótipo de extensão na ferramenta de programação por blocos;
4. Desenvolver o protótipo usando metodologia ágil: Serão programadas reuniões semanais para apresentação de resultados e ajustes de metas para a semana seguinte. Durante a semana os pontos indicados nas reuniões serão trabalhados gerando aperfeiçoamento do protótipo. No processo será gerado um diário de atividades.
5. Preparar material para publicação (site, relatórios, submissão de trabalhos no WICSI 2020 ou no CSBC ou melhor e no SIICUSP 2020).

## **Resultados esperados**

1. Site documentando o projeto;

2. Entrega do primeiro protótipo no fim de Janeiro;
3. Submissão de resumo ao WICSI no fim de Fevereiro;
4. Entrega de relatório intermediário no fim de Março;
5. Entrega da última versão da extensão em Agosto
6. Entrega do relatório final em Agosto
7. Submissão do resumo ao SIICUSP em Outubro.

## Cronograma de execução

	2019				2020							
	Set	Out	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago
T1												
T2												
T3												
T4												
T5												

T1 - Criação e atualização do site do projeto.

T2 - Preparação e submissão de artigos

T3 - Preparação e submissão de relatórios

T4 - Estudo das ferramentas

T5 - Criação do protótipo e ciclos de aperfeiçoamento

## Referências

BRADSKI, G. (2000) - The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

OpenCV (2019) - About OpenCV - <<https://opencv.org/about/>> Acessado em 26 de maio de 2019.

POKRESS, Shaileen Crawford; VEIGA, José Juan Dominguez (2013) - MIT App Inventor: Enabling Personal Mobile Computing - PProMoTo 2013 Proceedings (arXiv:1309.5509) October 2013.