# Computação Gráfica - IME – 2023

# Projeto 1 - Mapeamento Projetivo de Texturas

# Amanda Assis Lavinsky e Mayara Ribeiro Mendonça

1. **Exemplos gerados**

As imagens abaixo foram obtidas através de um mapeamento de projetivo de textura:

 

As imagens originais eram:

A person using a computer

Description automatically generated with medium confidence A picture containing text, tree, outdoor

Description automatically generated

As imagens utilizadas como texturas foram:

 

1. **Código Fonte**

O código abaixo foi usado para o Mapeamento Projetivo de Texturas. Além da opção de receber como entrada as coordenadas do quadrilátero, o programa também possui a função de selecionar os pontos com o clique do mouse.

|  |
| --- |
| import cv2  import numpy as np  bg=cv2.imread("outdoor.jpg")  txt=cv2.imread("militares.jpg")  (xb,yb,zb)=bg.shape  (xt,yt,zt)=txt.shape  bg = cv2.resize(bg,(800,int(800\*xb/yb)),interpolation=cv2.INTER\_AREA) # Resize para plotar imagem no espaço da tela  (xb,yb,zb)=bg.shape  PO=[(0,0,1),(xt,0,1),(xt,yt,1),(0,yt,1)]  PD=[]  # Selecionar vértices a partir do clique na imagem inicial  def select\_position():      def on\_click(event,x,y,flags,param):          if event==cv2.EVENT\_LBUTTONDOWN:              if len(PD)<4:                  PD.append((y,x,1))      cv2.imshow("image",bg)      cv2.namedWindow('image',cv2.WINDOW\_NORMAL)      cv2.setMouseCallback('image',on\_click)      cv2.waitKey(0)  select\_position()  # print("PD = ",PD)  # print("PO = ",PO)  # Calcular T tal que Ax=B e Tinv gera a transformação desejada  def transform():      A = np.array([[0 for i in range(12)] for j in range(12)])      B = np.array([0 for i in range(12)])      # Indices de lambda1=1      for i in range(3):          B[i]=PD[0][i]      # Indices de txy      for i in range(4):          for j in range(3):              A[3\*i][j]=PO[i][j]              A[3\*i+1][j+3]=PO[i][j]              A[3\*i+2][j+6]=PO[i][j]        # Indices de lambdax      for j in range(3):          for i in range(3):              A[3\*(j+1)+i][j+9]=-PD[j+1][i]      # print("matriz B\n", B)      # print("matriz A\n", A)      Ainv = np.linalg.inv(A)      x = np.dot(Ainv,B)      # print("matriz x\n", x)      T = np.array([[x[3\*j+i] for i in range(3)] for j in range(3)])      # print("matriz T\n", T)      Tinv = np.linalg.inv(T)      return(Tinv)  def generate\_picture(T):      for a in range(xb):          for b in range(yb):              v = (a,b,1)                j = T[1][0]\*v[0]+T[1][1]\*v[1]+T[1][2]\*v[2]              i = T[0][0]\*v[0]+T[0][1]\*v[1]+T[0][2]\*v[2]              k = T[2][0]\*v[0]+T[2][1]\*v[1]+T[2][2]\*v[2]              i=int(i/k)              j=int(j/k)              if i>=0 and j>=0 and i<xt and j<yt:                  bg[a][b]=txt[int(i)][int(j)]    T = transform()  generate\_picture(T)  cv2.imwrite("propaganda\_exercito.jpg",bg) |