Manual do utilizador

double CharacterRecognition(char* TargetImage,char* TemplateFileGroup[])	Correspondência de caracteres, suporta imagens BMP, e o valor de retorno é o número de sequência do arquivo de modelo correspondente à imagem de destino. Se o valor de retorno é 2, ele indica que a imagem corresponde ao modelo com o número de sequência 2 (a partir de zero). referência : TemplateFileGroup[]={ "0.txt", "1.txt", "2.txt", "3.txt", "4.txt", "5.txt", "6.txt", "7.txt", "8.txt", "9.txt" };
double CharacterRecognition1(char* TargetImage,char* TemplateFileGroup[])	Correspondência de caracteres, suporta imagens BMP, e o valor de retorno é o número de sequência do arquivo de modelo correspondente à imagem de destino. Se o valor de retorno é 2, ele indica que a imagem corresponde ao modelo com o número de sequência 2 (a partir de zero). referência : TemplateFileGroup[]={ "0.txt", "1.txt", "2.txt", "3.txt", "4.txt", "5.txt", "6.txt", "7.txt", "8.txt", "9.txt" };
void CodeEncoding(std::string input,char* output, int width,int height, int margin, int eccLevel, int stride_bytes, int comp,int a)	Código QR e codificação de código de barras. Entrada é a string a ser codificada, e saída é o nome do arquivo da imagem gerada do código QR. margem: A margem em torno do código de barras ecc: Nível de correcção de erros, [0-8] a=1：AZTEC a=2：CODABAR a=3：CODE_39 a=4：CODE_93 a=5：CODE_128 a=6：DATA_MATRIX a=7：EAN_8 a=8：EAN_13

	a=9：ITF a=10：MAXICODE a=11：PDF_417 a=12：QR_CODE a=13：RSS_14 a=14：RSS_EXPANDED a=15：UPC_A a=16：UPC_E a=17：UPC_EAN_EXTENSION Referência：margin=10，eccLevel=-1，stride_bytes=0，comp=1。
std::string CodeDecoding(char* input,int req_comp,int a)	Decodificação de códigos QR e códigos de barras. Entrada é o nome do arquivo da imagem de código QR de entrada e retorna o resultado de decodificação. a=1：Lum a=2：RGB a=3：BGR a=4：RGBX a=5：XRGB a=6：BGRX a=7：XBGR Referência：req_comp=4，a=4。