

---

PTDC-AUR-66476-2006:

**“Contributos para o Projecto de Conservação do Património Arquitectónico:  
Metodologia documental baseada na fotogrametria digital e na digitalização laser 3D  
terrestre”**

**Investigador responsável: Professor José Aguiar**

---

**RELATÓRIO**

**Método Prático de Levantamento Arquitectónico  
- Fotogrametria de baixo custo -**

## Índice

	Pag.
1. Introdução	2
2. Descrição geométrica	3
2.1. A rectificação	3
2.2. A determinação do ângulo entre fachadas	8
3. Descrição da ferramenta informática	11
4. Aplicações práticas	12
5. Conclusões e desenvolvimentos	19
6. Bibliografia	20
Anexo 1 – Código autolisp da rotina para rectificação	21
Anexo 2 – Código autolisp da rotina para determinação do ângulo entre fachadas	29
Anexo 3 – Código para alteração do ficheiro acad.mnu	33

## 1. Introdução

Para elaborar o levantamento de um edifício várias abordagens podem ser adoptadas. Uma são mais práticas e expeditas e outras mais morosas e rigorosas, devendo sempre considerar-se a adequação dos meios aos objectivos.

O método aqui apresentado e desenvolvido, que em certo sentido se pode definir como um método de fotogrametria elementar, aplica-se sobretudo ao levantamento de edifícios de fachadas aproximadamente planas. Tem como base operativa duas operações base que recorrem às propriedades da geometria projectiva aplicadas a desenhos elaborados, em ambiente de AutoCAD, sobre fotografias digitais.

A primeira operação consiste na rectificação de desenho vectorial produzido sobre fotografia para a produção de levantamento de fachadas, sendo que o que se pretende é um alçado e não uma foto rectificada. Note-se que numa foto rectificada, embora se disponha da informação radiométrica, apenas se podem efectuar medições num plano de fachada, o plano rectificado. Em planos paralelos a esse não é possível, a menos que se altere a escala. Por outro lado, uma foto rectificada, mesmo que se trate de um fotomosaico, apresenta sempre algumas distorções próprias da perspectiva que, no presente caso, se pretende evitar. Embora se reconheça que uma foto rectificada possa ser útil, por exemplo, para a marcação e registo de patologias que se podem identificar pela textura da fotografia, à semelhança do que é feito na Carta de Risco da Direcção Geral de Edifícios e Monumentos Nacionais, um alçado é sempre necessário para informar um processo de projecto, mesmo que seja complementado por fotografias, rectificadas ou não. Contudo, o alçado produzido pode servir de base métrica para a rectificação fotográfica. Isto é válido para um edifício isolado e para um conjunto de edifícios de uma rua.

A segunda operação consiste em elaborar, com o mínimo possível de trabalho de campo, o levantamento da planta do contorno de um edifício mais ou menos isolado através da determinação dos ângulos entre os planos das várias fachadas. Efectivamente, esta segunda operação do método não é aplicável no caso do edifício a registar apenas apresentar uma fachada, isto é, se o edifício estiver confinado entre edifícios, por exemplo numa rua.

Sendo aplicável a operação, pretende-se obter um polígono que delimita o edifício e que pode enquadrar um posterior levantamento, através de métodos clássicos directos, do interior do edifício. Deste modo pretende-se eliminar a necessidade de marcação de uma poligonal exterior ao edifício para apoiar um processo de triangulação.

Não se pretende defender a eficácia deste método relativamente qualquer outro. Pretende-se apontar que pode funcionar em complemento de outros métodos. Trata-se seguramente de um método que apresenta vantagens e inconvenientes.

Para tornar operativo o método programaram-se duas rotinas em autolisp que foram integradas num *plug-in* para o software AutoCAD.

## 2. Descrição geométrica

### 2.1. A rectificação

A operação de rectificação baseia-se no teorema da geometria projectiva segundo o qual “*quaisquer quatro pontos num plano  $\pi$ , não sendo três colineares, são projectivos com quaisquer quatro pontos num plano  $\pi'$ , não sendo três colineares*”. Este teorema não exclui que os dois planos coincidam e traduz que dada uma figura inicial num plano  $\pi$  e quatro pontos num plano  $\pi'$  que correspondam a quatro pontos conhecidos da figura inicial, é possível estabelecer os parâmetros que permitem efectuar a transformação da figura inicial, do plano  $\pi$ , na figura final, no plano  $\pi'$  (mesmo que os dois planos sejam apenas um).

Na prática, na generalidade dos casos, as coordenadas dos pontos do plano  $\pi'$  resultam da *medição in-situ* de uma largura e uma altura na fachada do edifício, o que permite desenhar um rectângulo de referência. Os pontos homólogos dos vértices do rectângulo de referência, ou seja, os pontos do plano  $\pi$ , são pontos que se identificam sobre a fotografia.

A figura seguinte permite ilustrar o princípio.

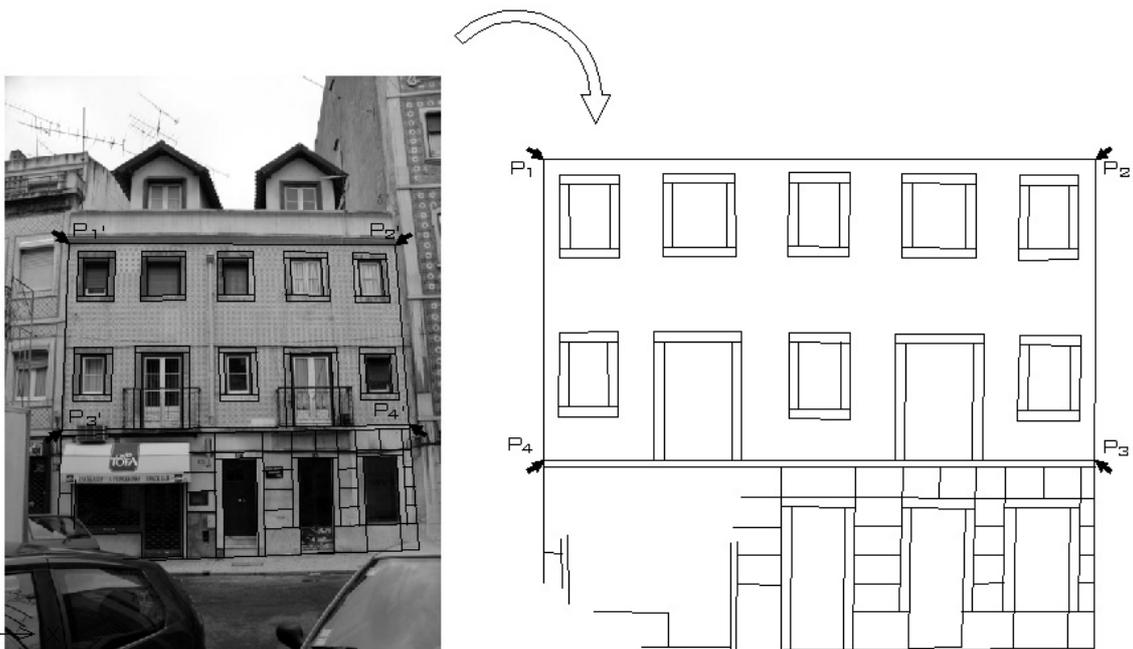


Fig. 1

Esta transformação projectiva pode ser traduzida analiticamente por:

$$X_u = \frac{e_1 x_n + f_1 y_n + g_1}{e_0 x_n + f_0 y_n + 1}$$

$$Y_u = \frac{e_2 x_n + f_2 y_n + g_2}{e_0 x_n + f_0 y_n + 1}$$

Em que:

$X_u$  e  $Y_u$  são as coordenadas do ponto  $P_u$  a determinar,

$x_n$  e  $y_n$  são as coordenadas do ponto  $P_n$  conhecido e correspondente ao ponto  $P_u$ ,

$e_0, e_1, e_2, f_0, f_1, f_2, g_1, g_2$  são os oito parâmetros da transformação projectiva.

Deduz-se das expressões que, tal como já foi afirmado, para determinar os parâmetros da transformação são necessárias as coordenadas de quatro pontos de referência,  $P_i$ , no plano do desenho a obter, e de quatro pontos de referência,  $P'_i$ , no plano do desenho elaborado, correspondentes aos anteriores.

É de notar que a determinação dos oito parâmetros da transformação, tratada na sua generalidade, é pouco prática pois consiste em resolver um sistema não linear de oito equações a oito incógnitas, pelo que se adoptou uma estratégia que tornou mais expedita a operação. Se três das coordenadas dos pontos de referência  $P_i$  e três das coordenadas dos pontos de referência  $P'_i$  forem iguais a 0, então a resolução do sistema torna-se bastante simples. Como, em geral, esta condição não se cumpre à partida, considerou-se um conjunto de transformações geométricas que alteram as coordenadas dos pontos iniciais de modo a simplificar a determinação dos parâmetros da transformação projectiva, e um outro conjunto de transformações geométricas que trazem os pontos transformados à posição que se pretende.

A sequência de transformações consistiu em:

a) Aplicar uma translacção dos pontos de referência  $P_i$  ( $i = \{1,2,3,4\}$ ), de coordenadas  $X_i$  e  $Y_i$ , segundo o vector  $\overrightarrow{P_4O}$  de modo a obter os pontos  $P_{it}$ , de coordenadas  $X_{it}$  e  $Y_{it}$ , de acordo com as seguintes expressões:

$$X_{it} = X_i - X_4$$

$$Y_{it} = Y_i - Y_4$$

b) Aplicar uma translacção dos pontos de referência  $P'_i$  ( $i = \{1,2,3,4\}$ ), de coordenadas  $x_i$  e  $y_i$ , e aos pontos  $P_n$ , de coordenadas  $x_n$  e  $y_n$ , da figura  $ff$  a transformar, segundo o vector  $\overrightarrow{P'_1O}$  de modo a obter os pontos  $P'_{it}$  de coordenadas  $x_{it}$  e  $y_{it}$ , e os pontos  $P_{nt}$ , da figura  $ff'$ , de coordenadas  $x_{nt}$  e  $y_{nt}$ , de acordo com as seguintes expressões:

$$x_{it} = x_i - x_1$$

$$y_{it} = y_i - y_1$$

$$x_{nt} = x_n - x_1$$

$$y_{nt} = y_n - y_1$$

c) Aplicar uma rotação dos pontos  $P_{it}$  em torno da origem de modo a alinhar o vector  $\overrightarrow{P_{4t}P_{3t}}$  com o eixo dos  $xx$ , obtendo os pontos  $P_{itr}$  de coordenadas  $X_{itr}$  e  $Y_{itr}$ , de acordo com as seguintes expressões:

$$X_{itr} = X_{it} \cos \alpha + Y_{it} \sin \alpha$$

$$Y_{itr} = -X_{it} \sin \alpha + Y_{it} \cos \alpha$$

em que:

$$\cos \alpha = \frac{X_{3t}}{\sqrt{X_{3t}^2 + Y_{3t}^2}}$$

$$\sin \alpha = -\frac{Y_{3t}}{\sqrt{X_{3t}^2 + Y_{3t}^2}}$$

d) Aplicar uma rotação aos pontos  $P'_{it}$  e aos pontos  $P_{nt}$ , da figura  $ff_t$ , em torno da origem de modo a alinhar o vector  $\overrightarrow{P'_{1t}P'_{2t}}$  com o eixo dos  $xx$ , obtendo os pontos  $P'_{itr}$  de coordenadas  $x_{itr}$  e  $y_{itr}$  e os pontos  $P_{ntr}$ , da figura  $ff_{tr}$ , de coordenadas  $x_{ntr}$  e  $y_{ntr}$ , de acordo com as seguintes expressões:

$$x_{itr} = x_{it} \cos \beta + y_{it} \sin \beta$$

$$y_{itr} = -x_{it} \sin \beta + y_{it} \cos \beta$$

$$x_{ntr} = x_{nt} \cos \alpha + y_{nt} \sin \beta$$

$$y_{ntr} = -x_{nt} \sin \beta + y_{nt} \cos \beta$$

em que:

$$\cos \beta = \frac{x_{2t}}{\sqrt{x_{2t}^2 + y_{2t}^2}}$$

$$\sin \beta = -\frac{y_{2t}}{\sqrt{x_{2t}^2 + y_{2t}^2}}$$

e) Determinar os parâmetros  $e'_0, e'_1, e'_2, f'_0, f'_1, f'_2, g'_1, g'_2$  da transformação projectiva que relaciona os quatro pontos  $P_{itr}$  e os quatro pontos correspondentes  $P'_{itr}$ , que, em função das novas coordenadas dos pontos de referência podem ser obtidos de acordo com as seguintes expressões:

$$g'_1 = X_{1tr}$$

$$g'_2 = Y_{1tr}$$

$$f'_2 = \frac{Y_{1tr}x_{4tr} - Y_{1tr}x_{3tr}}{x_{3tr}y_{4tr} - y_{3tr}x_{4tr}}$$

$$e'_2 = \frac{Y_{1tr}y_{3tr} - Y_{1tr}y_{4tr}}{x_{3tr}y_{4tr} - y_{3tr}x_{4tr}}$$

$$e'_1 = \frac{e'_2 X_{2tr}}{Y_{2tr}} + \frac{X_{2tr}(Y_{1tr} - Y_{2tr})}{x_{2tr}Y_{2tr}} - \frac{X_{1tr} - X_{2tr}}{x_{2tr}}$$

$$f'_1 = -\frac{e'_1 x_{4tr} + X_{1tr}}{y_{4tr}}$$

$$e'_0 = \frac{e'_2}{Y_{2tr}} + \frac{Y_{1tr} - Y_{2tr}}{x_{2tr}Y_{2tr}}$$

$$f'_0 = \frac{e'_1 x_{3tr} + f'_1 y_{3tr} + X_{1tr} - e'_0 x_{3tr}X_{3tr} - X_{3tr}}{y_{3tr}X_{3tr}}$$

f) Aplicar a transformação projectiva aos pontos  $P_{ntr}$  da figura  $ff_{tr}$ , de coordenadas  $x_{ntr}$  e  $y_{ntr}$ , de modo a obter os pontos  $P_{utr}$  da figura  $FF_{tr}$ , de coordenadas  $X_{utr}$  e  $Y_{utr}$ , de acordo com as seguintes expressões:

$$X_{utr} = \frac{e'_1 x_{ntr} + f'_1 y_{ntr} + g'_1}{e'_0 x_{ntr} + f'_0 y_{ntr} + 1}$$

$$Y_{utr} = \frac{e'_2 x_{ntr} + f'_2 y_{ntr} + g'_2}{e'_0 x_{ntr} + f'_0 y_{ntr} + 1}$$

g) Aplicar uma rotação aos pontos  $P_{utr}$ , da figura  $FF_{tr}$ , de coordenadas  $X_{utr}$  e  $Y_{utr}$ , em torno da origem com a mesma amplitude e sentido contrário à aplicada em c) obtendo os pontos  $P_{ut}$  da figura  $FF_t$ , de coordenadas  $X_{ut}$  e  $Y_{ut}$ , de acordo com as seguintes expressões:

$$X_{ut} = X_{utr} \cos(-\alpha) + Y_{utr} \sin(-\alpha)$$

$$Y_{ut} = -X_{utr} \sin(-\alpha) + Y_{utr} \cos(-\alpha)$$

em que:

$$\cos(-\alpha) = \cos \alpha = \frac{X_{3t}}{\sqrt{X_{3t}^2 + Y_{3t}^2}}$$

$$\sin(-\alpha) = -\sin \alpha = \frac{Y_{3t}}{\sqrt{X_{3t}^2 + Y_{3t}^2}}$$

h) Aplicar uma translacção aos pontos  $P_{ut}$  da figura  $FF_t$ , de coordenadas  $X_{ut}$  e  $Y_{ut}$ , segundo o vector  $\overrightarrow{OP_4}$  obtendo os pontos  $P_u$  da figura  $FF$ , de coordenadas  $X_u$  e  $Y_u$ , de acordo com as seguintes expressões:

$$X_u = X_{ut} + X_4$$

$$Y_u = Y_{ut} + Y_4$$

Esta seqüência de operações vem esquematizada na figura seguinte.

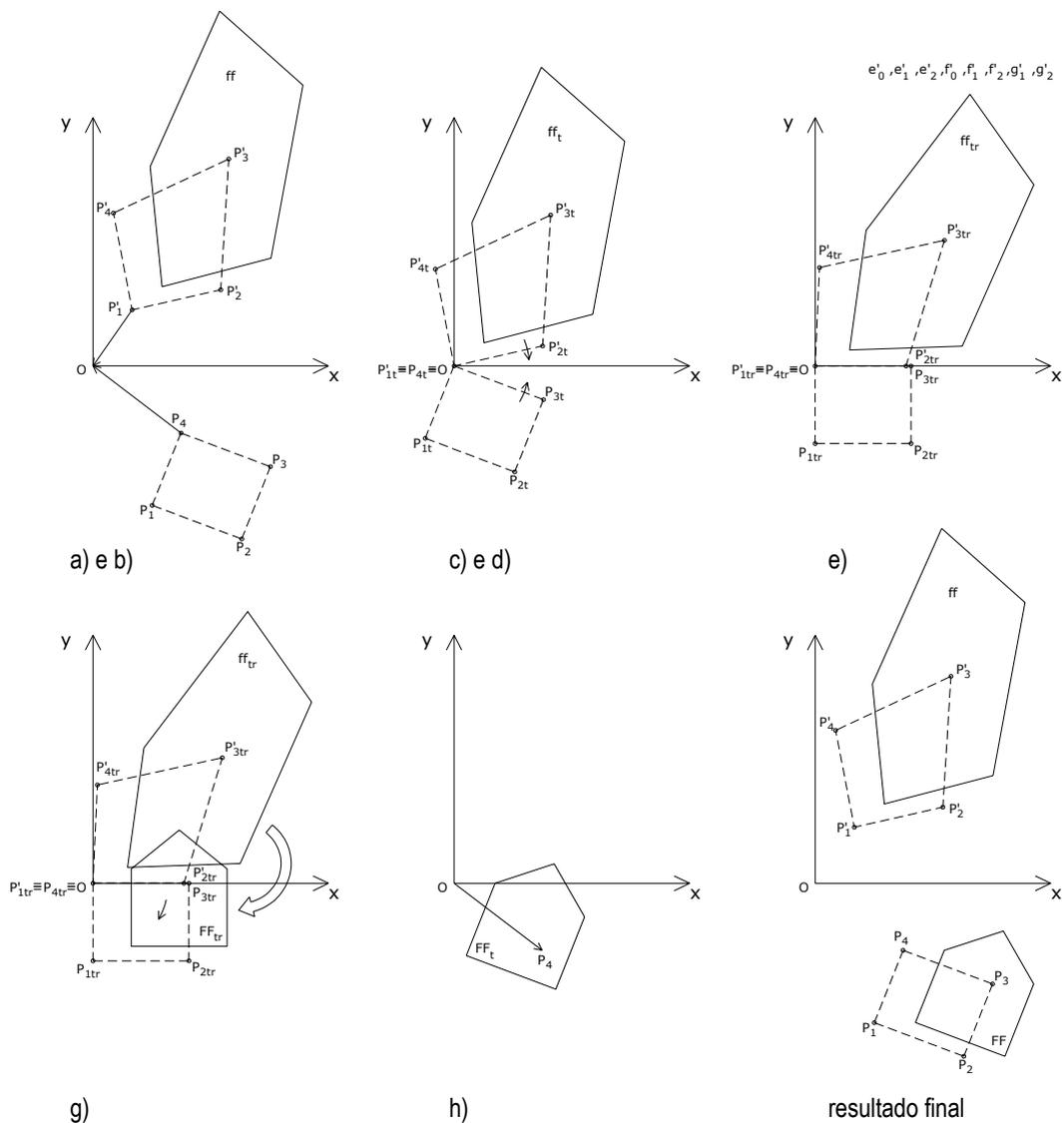


Fig. 2



Seja  $D_d K = [OB] = R$ , em que  $K$  é um factor de escala determinado do seguinte modo:

$$K = \frac{V_C}{V_{C'}}$$

Seja  $[ON]$  a distância, medida sobre a foto, entre as verticais  $V_C$  e  $V_d$ .

Seja  $A$  um ponto qualquer na fachada (na prática este ponto não é identificado pelo operador; assume-se este ponto como o cruzamento das diagonais do rectângulo correspondente à fachada direita).

Seja  $M$  o ponto, na foto, homólogo ao ponto  $A$  (na prática este ponto não é identificado pelo operador; assume-se este ponto como o cruzamento das diagonais do trapézio homólogo do rectângulo correspondente à fachada direita).

Fazendo corresponder o eixo dos  $xx$  ao plano da foto, é indeterminado o ângulo  $\alpha$  que este plano faz com o plano da fachada direita. Contudo é sabido que as rectas projectantes  $AM$  e  $BN$  se cruzam no centro de projecção  $P$ .

Sejam as coordenadas dos pontos  $A$ ,  $B$ ,  $M$ ,  $N$  e  $P$  as seguintes:

$$A = (r \cos \alpha; r \sin \alpha)$$

$$B = (R \cos \alpha; R \sin \alpha)$$

$$M = (d; 0)$$

$$N = (D; 0)$$

$$P = (X_p; Y_p)$$

Os alinhamentos  $AM.P$  e  $BN.P$  podem ser traduzidos pelas seguintes igualdades:

$$\begin{vmatrix} r \cos \alpha & d & X_p \\ r \sin \alpha & 0 & Y_p \\ 1 & 1 & 1 \end{vmatrix} = 0$$

$$\begin{vmatrix} R \cos \alpha & D & X_p \\ R \sin \alpha & 0 & Y_p \\ 1 & 1 & 1 \end{vmatrix} = 0$$

Estas igualdades permitem definir um sistema de duas equações a duas incógnitas que pode ser resolvido em ordem a  $X_p$  e  $Y_p$ , obtendo-se os seguintes resultados:

$$Y_P = \frac{Rr(d-D)}{Rd-rD} \sin \alpha$$

$$X_P = \frac{Rr(d-D)}{Rd-rD} \cos \alpha + \frac{Dd(R-r)}{Rd-rD}$$

Estes expressam as equações paramétricas de uma circunferência  $[c]$  de centro  $C = \left( \frac{Dd(R-r)}{Rd-rD}; 0 \right)$  e raio

$$R_{[c]} = \left| \frac{Rr(d-D)}{Rd-rD} \right|.$$

Procedendo de igual modo para a fachada à esquerda na foto determina-se uma circunferência  $[e]$  de centro  $E$  e raio  $R_{[e]}$ .

O centro de projecções  $P$  que permite reconstituir simultaneamente a geometria da projecção dos planos das duas fachadas é um dos pontos de intersecção das circunferências  $[c]$  e  $[e]$ , como indicado na figura seguinte. Deste modo determina-se o ângulo  $\pi$  que fazem as duas fachadas referido às medidas  $D_e K$  e  $D_d K$ . O factor de escala pode ser eliminado dividindo ambas as medidas por  $K$ .

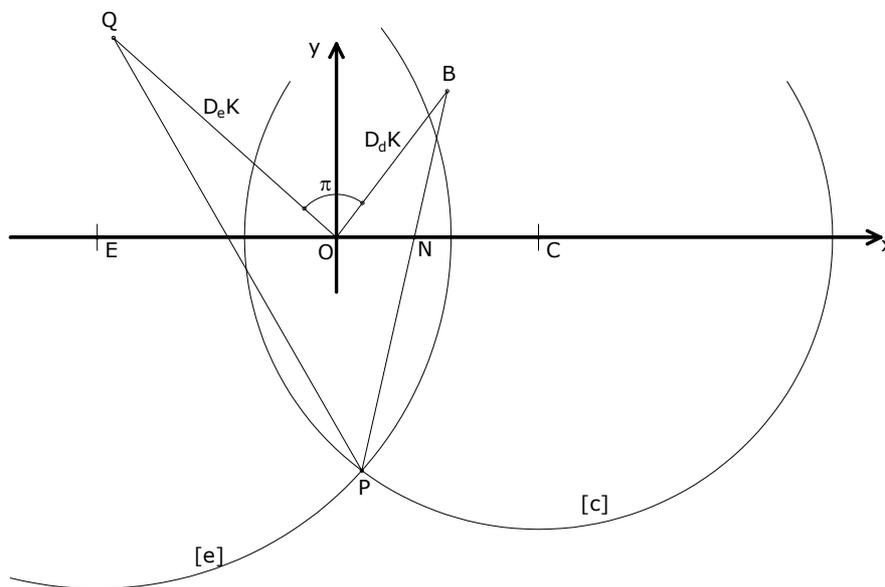


Fig. 5

### 3. Descrição da ferramenta informática

Com base na geometria descrita no ponto anterior, foram programadas duas rotinas em Autolisp<sup>1</sup> que foram integradas num *plug-in* para o *software* AutoCAD. Este, corresponde à adição de um menu no *layout* do AutoCAD e de uma barra de ferramentas que agregam as funções definidas nas rotinas, ambos com a designação Ferr\_Lev (Ferramentas de levantamento). Através do menu carregam-se as aplicações.

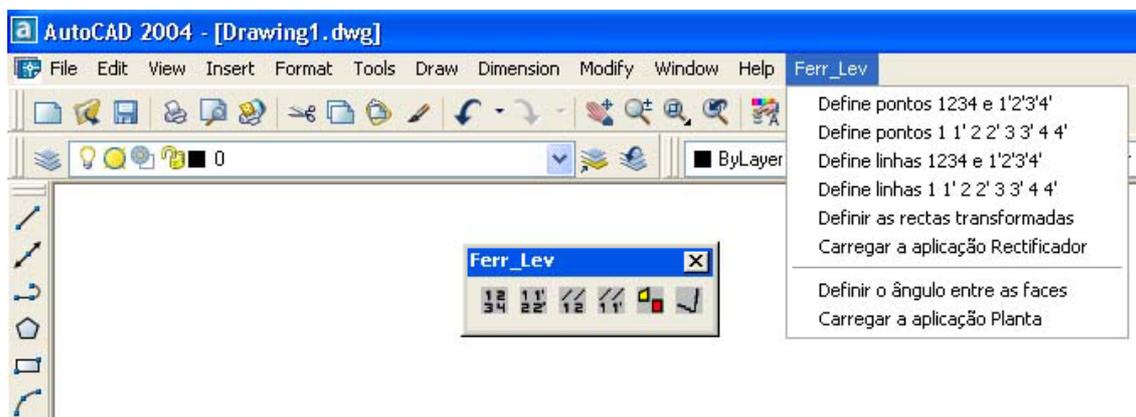


Fig. 6

A execução das rotinas pode ser invocada através dos ícones ou através do menu.

A ordem pela qual aparecem os ícones corresponde à ordem pela qual aparecem as descrições das operações no menu.

As funções correspondentes aos ícones são as seguintes:

-  Pedir a selecção de quatro pontos de referência no plano da imagem, quatro pontos homólogos no plano do desenho a obter, e os elementos a rectificar.
-  Pedir a selecção dos pontos homólogos através de pares, e os elementos a rectificar.
-  Pedir a selecção de quatro rectas de referência no plano da imagem, quatro rectas homólogas no plano do desenho a obter, e os elementos a rectificar.
-  Pedir a selecção das rectas homólogas através de pares, e os elementos a rectificar.
-  Pedir os elementos a rectificar quando os elementos de referência homólogos já foram definidos em operações anteriores.
-  Pedir a selecção de elementos para a execução da rotina de determinação do ângulo entre fachadas.

<sup>1</sup> Em anexo é dado o código em Autolisp das duas rotinas e o código que permite a alteração do ficheiro acad.mnu de modo a instalar o *plug-in*.

#### 4. Aplicações práticas

Neste ponto apresentam-se e descrevem-se duas aplicações práticas.

A primeira consistiu no levantamento da fachada principal da Igreja de São João da Talha no Concelho de Loures.

A segunda consistiu no levantamento da fachada principal e da planta do contorno da Igreja de Santa Iria da Azóia, também no Concelho de Loures.

Em ambos os casos foi utilizada uma máquina fotográfica da marca Olympus, modelo E-500, com resolução de 8 mega-pixels, com lente zoom 14-45mm.

Na primeira aplicação, em que apenas se efectuou o levantamento da fachada, podemos considerar dois passos no processo de trabalho.

O primeiro passo do processo consiste em trabalho de campo e tem dois momentos.

Em primeiro lugar, tiram-se um conjunto de fotos digitais, neste caso apenas 5 fotografias, em que pelo menos duas, de ângulos complementares, enquadrem globalmente a fachada.

Em segundo lugar, devem registar-se as coordenadas de quatro pontos, o que na prática, na maior parte dos casos, consiste em medir uma largura e uma altura que devem ser o mais abrangentes possível.

Neste caso as medidas foram obtidas com fita métrica tendo sido a medida horizontal mais abrangente que a vertical por questões logísticas. Esta questão pode ser ultrapassada se se utilizar um distanciómetro *laser*.



Fig. 7

O segundo passo do processo consiste em trabalho de gabinete.

Inserem-se as fotos digitais no ambiente de trabalho do *software* AutoCAD e procede-se à sua vectorização. As fotos que enquadram globalmente a fachada servem para produzir um desenho geral cujos vértices servem como pontos de referência para o registo de desenhos parcelares elaborados sobre fotos de pormenor.



Fig. 8

Estes elementos foram montados, através duma primeira rectificação, num desenho de conjunto (Fig. 9).

Neste desenho não é importante, ainda, a escala nem a dimensão. Pode-se assumir, por exemplo, um rectângulo que se identifique na fachada. Neste caso, assumiu-se que os pontos  $P'_A$ ,  $P'_B$ ,  $P'_C$  e  $P'_D$  correspondiam a um rectângulo. A vantagem de considerar um rectângulo está relacionada com uma primeira percepção dos desvios em relação às direcções horizontal e vertical que pode ajudar a controlar melhor o desenho que vai sendo produzido. Note-se no entanto que, do ponto de vista teórico, não é necessário considerar um rectângulo, isto é, qualquer quadrilátero serve.

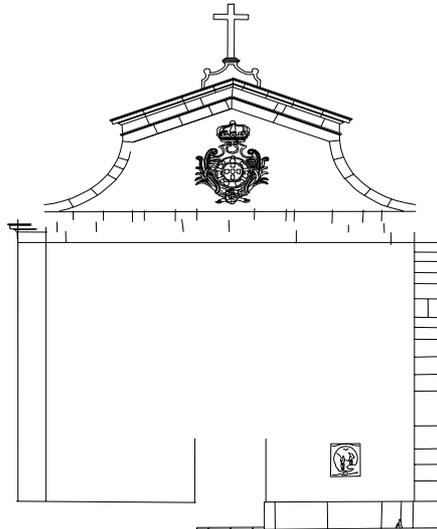


Fig. 9

Prosseguindo com as operações, o desenho vai sendo completado.

Por fim, com base nas medidas tiradas em campo, realiza-se a operação de rectificação que coloca o desenho na escala pretendida de modo a poderem realizar-se medições sobre este.



Fig. 10

Sobre o desenho final (fig. 11) foram realizadas medições, apenas na horizontal, que foram comparadas com medições efectuadas *in-situ*.

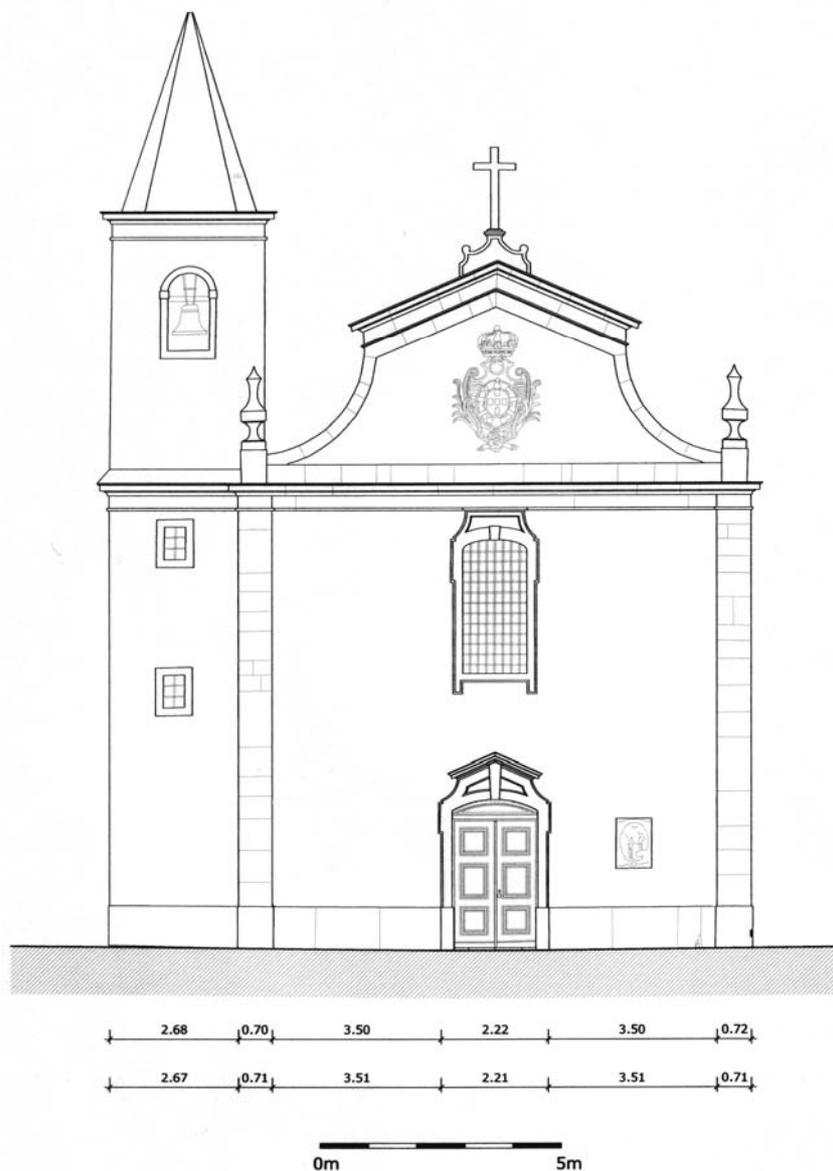


Fig. 11

Na figura apresentam-se duas linhas de cotagem. A linha superior corresponde a uma medição efectuada sobre o desenho produzido através deste método. A linha inferior corresponde a medições efectuadas *in-situ*. Verifica-se, pela comparação entre as medidas que figuram nas duas linhas, que as diferenças obtidas rondam  $\pm 1$ cm.

Como nota final, indica-se que este alçado demorou 8h a 10h a ser produzido, incluindo trabalho de campo.

Na segunda aplicação, em que se efectuou o levantamento da fachada principal e o levantamento em planta do contorno do edifício, no que concerne ao levantamento da fachada, houve o cuidado de utilizar maior quantidade de informação fotográfica o que permitiu maior rigor na definição de detalhes de molduras, vãos, cimalkas, estereotomias, etc.

Neste caso utilizaram-se 9 fotos para o levantamento da fachada tendo sido os procedimentos semelhantes aos adoptados na primeira aplicação prática, embora com um resultado mais satisfatório do ponto de vista do detalhe.

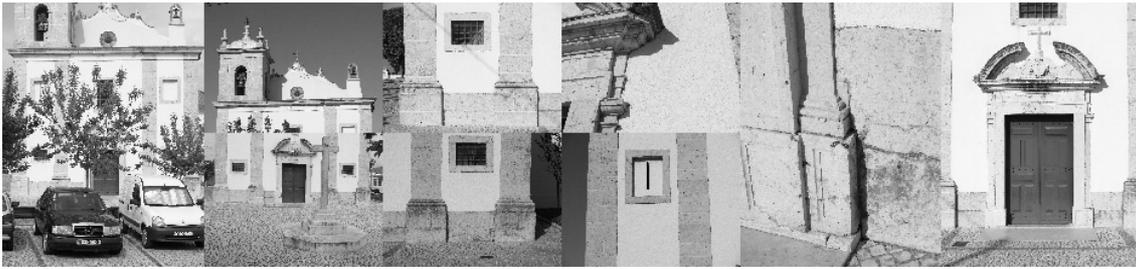


Fig. 12

Deve apontar-se que a dificuldade de restituição aumenta se os elementos a representar não forem planos o que se traduzirá num acréscimo de horas de trabalho.

Na prática, isto significa que o desenho deve ser produzido atendendo às regras da perspectiva e tendo por base informação geométrica que se pode deduzir da foto.

Os elementos exteriores ao plano a rectificar devem ser projectados neste de modo a proceder-se depois à sua rectificação.



Fig. 13

Indica-se que este alçado demorou mais tempo a produzir que o anterior uma vez que existem vários planos com distintas profundidades bem como uma torre e torreões de configuração não poliédrica e não plana.



Fig.14

Neste alçado não foram feitas comparações entre medições efectuadas sobre o desenho e medições efectuadas em campo.

Para a definição da planta do contorno do edifício foram utilizadas 10 fotos que, tal como já foi referido, foram tiradas com o eixo da objectiva na horizontal. Para o efeito utilizou-se um tripé.

À semelhança do que acontece com a rectificação, também para a realização desta operação podemos considerar dois passos no processo de trabalho.

O primeiro passo corresponde a trabalho de campo e divide-se em duas etapas.

Na primeira etapa tiram-se um conjunto de fotos das várias esquinas do edifício. Estas fotos, tal como já foi anteriormente referido, devem ser tiradas com o eixo da objectiva na horizontal. Para o efeito pode utilizar-se um tripé. Neste caso foram tiradas 10 fotos.

Na segunda etapa devem medir-se as larguras totais de todos os planos de fachada do edifício, e uma altura ( a altura pode extrair-se da rectificação da fachada).

O segundo passo do processo corresponde a trabalho de gabinete e também se pode dividir em duas etapas.

Na primeira etapa produz-se um ficheiro de AutoCAD em que se representam rectângulos correspondentes aos vários planos das várias fachadas na escala pretendida. De seguida inserem-se as fotos no mesmo ficheiro e representam-se sobre estas os trapézios homólogos dos rectângulos.

Produzidos estes elementos pode proceder-se à determinação dos ângulos entre fachadas de acordo com o descrito em 2.2.

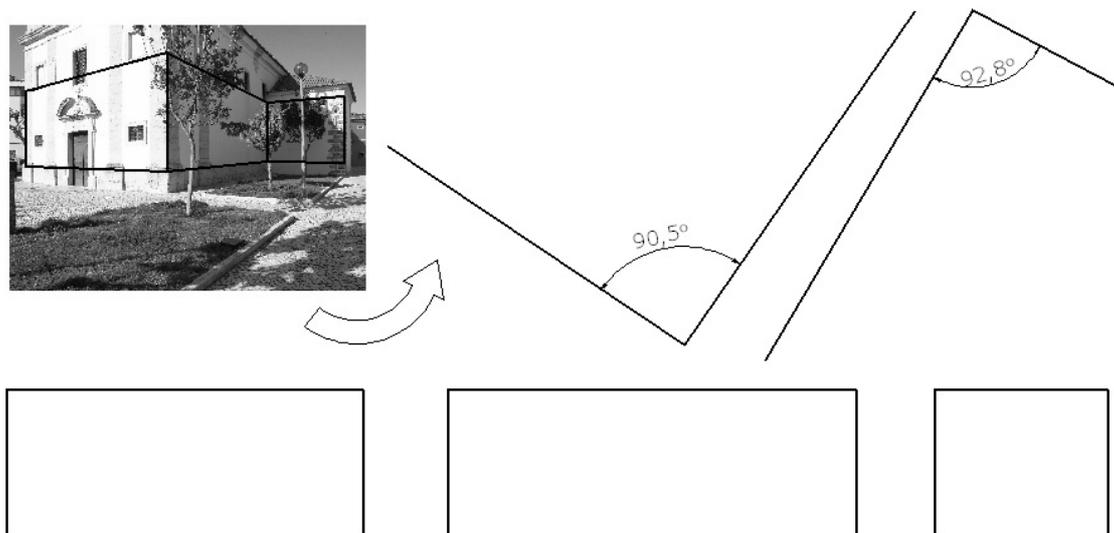


Fig. 15

Trata-se de uma operação que é realizada para cada par de fachadas.

Sobre as fotos a altura de referência pode ser propagada através de traçados de simples construção atendendo ao carácter perspéctico da imagem.

Os vários ângulos obtidos e respectivas posições relativas entre planos de fachadas obtidos em planta devem agora ser integrados num único desenho (fig. 16).

A produção deste desenho é sequencial e por isso é natural que suceda um acumular de erros a cada operação que se realiza. Neste caso, que não corresponde a um edifício cujo perímetro seja fechado, verifica-se que os planos de duas fachadas, que deveriam ser aproximadamente paralelos, apresentam um grau de obliquidade de 1.4°. Este dado só por si não significa nada nem dá nenhuma indicação sobre o rigor global do polígono desenhado. Para se fazer uma verificação do rigor da planta produzida por este meio deveria proceder-se uma comparação com uma planta efectuado através de um processo já testado, por exemplo através de topografia.

Finaliza-se indicando que esta planta demorou cerca de 6h a 8h a ser produzida, incluindo o trabalho de campo.

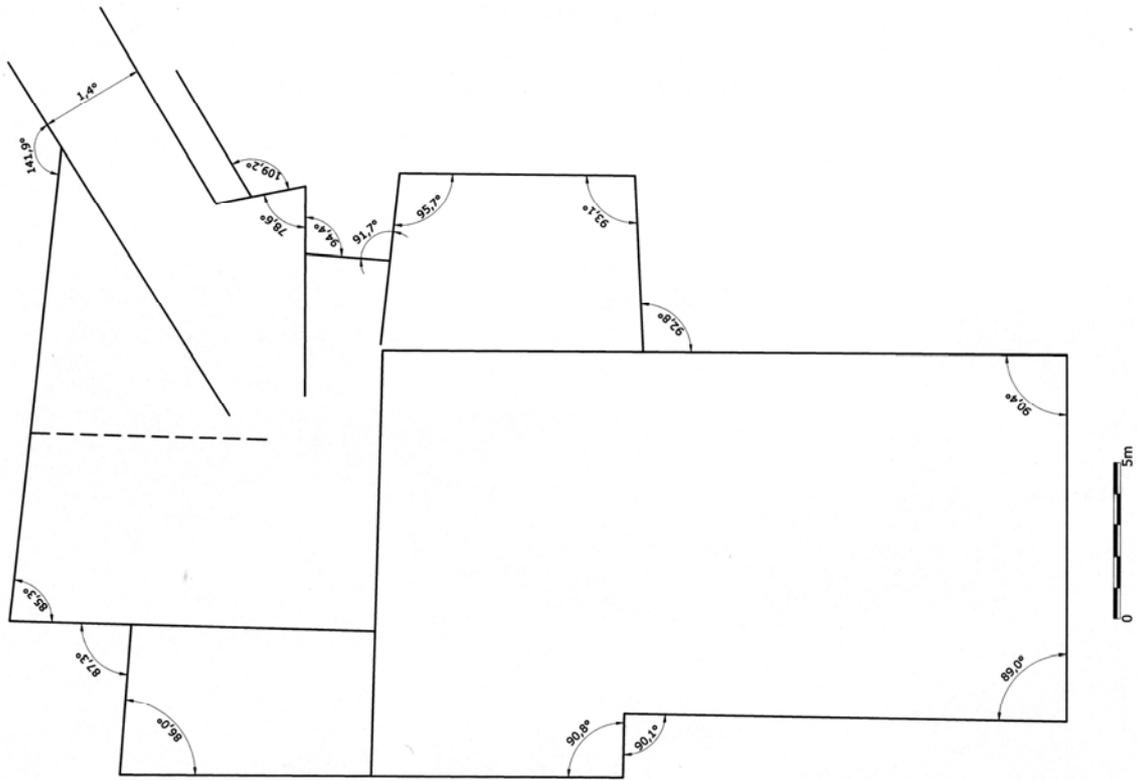


Fig. 16

## 5. Conclusões e desenvolvimentos

É ainda necessário proceder a mais aplicações do método para aferir com mais rigor as margens de erro que se pratica ao adoptar este método.

Por exemplo, é de interesse, comparar o levantamento fotogramétrico estereoscópico de uma fachada com um levantamento produzido através do processo descrito.

Contudo, pelo valor das diferenças verificadas considera-se que a produção de levantamento de fachadas através deste método, que é bastante mais célere que o método tradicional, pode considerar-se satisfatório para a generalidade das operações de Projecto passíveis de representação à escala 1/100 ou 1/50.

Embora não tenha sido pessoalmente testado, este método já foi utilizado com sucesso para medir áreas de patologias sobre fachadas.

Como perspectiva de desenvolvimento futuro, considera-se o desenvolvimento de uma ferramenta informática que permita a importação de textura da fotografia, de forma expedita, para o desenho de alçado produzido através da rectificação do desenho vectorial.

No que concerne à segunda parte do método, os dados obtidos através do exemplo apresentado permitem supor que o método pode ser utilizado na prática. Contudo, deve testar-se o método num edifício completamente isolado para que se possa verificar a acumulação final de erro. Em função dessa verificação poderá deduzir-se um processo que permita reduzir o erro.

Para além desta abordagem, é também conveniente comparar uma planta produzida através deste método com uma planta produzida através doutro método, por exemplo topográfico. Deste modo também se poderá avaliar a margem de erro que se comete.

Sendo um método que se apoia sobretudo em fotografia apresenta a vantagem da economia de tempo em campo que se traduz pela não necessidade de efectuar operações de triangulação.

Como mais valia da ferramenta informática salienta-se o facto de se poder manipular desenho vectorial independentemente da sua origem a foto ou não. A figura seguinte é disso exemplo.

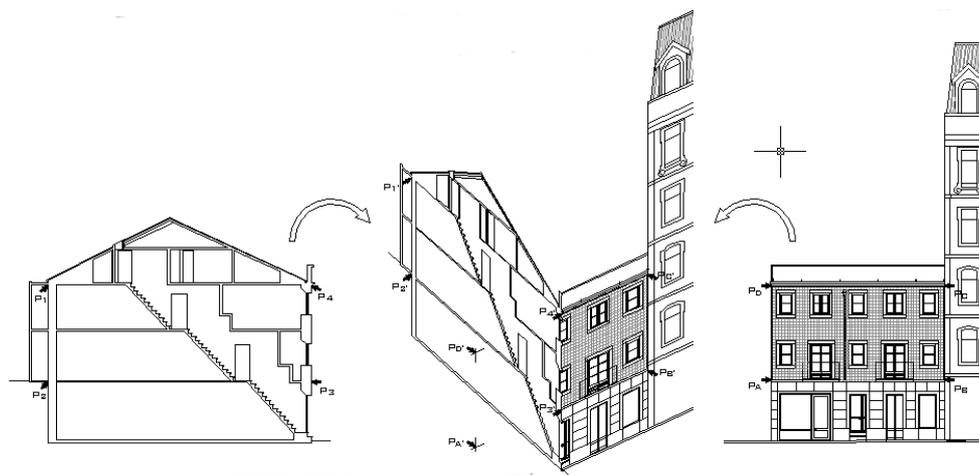


Fig. 17

Como limitação da ferramenta informática, no que concerne à rectificação, apresenta-se o facto de esta apenas operar sobre segmentos de recta.

## 6. Bibliografia

CARVALHO, José Maria Lobo de; "The Heritage at Risk", Historical Constructions 2001, Paulo Lourenço e Pere Roca (Eds.), Universidade do Minho, Guimarães, 2001, pp.277-282.

FEIFFER, Cesare – Il progetto di conservazione. Milão: Franco Angeli Libri s.r.l., 1989. 595p. ISBN 88-204-3055-X

MAESTRI, Diego; DOCCI, Mario – Manuale di rilevamento architettonico e urbano. 8ª edição. Roma: Editori Laterza, 2005. 343p. ISBN 88-420-4341-9

MATEUS, Luís; "Rectificação de desenho vectorial", Novas perspectivas em Sistemas e Tecnologias de Informação (CD), edições Universidade Fernando Pessoa, 2007, ISBN 978-972-8830-88-5

MIKHAIL, Edward M.; BETHEL, James S. & MCGLONE, J. Chris – Introduction to Modern Photogrammetry. Nova Iorque: John Wiley & Sons, 2001. 479p. ISBN 0-471-30924-9

AutoLISP Programmers Reference - Autocad release 12, Neuchatel: Autodesk BV, 1992. 251p. ISBN 2-88447-004-2

## Anexo 1 – Código autolisp da rotina para rectificação

```
*****
;
; *
; *
; * Rotina para resolver transformações projectivas sobre *
; * desenhos produzidos exclusivamente com linhas rectas *
; *
; *
; *****
;
; *****
;
; *
; * Portugal - Lisboa, Dezembro, 2007 *
; *
; *
; * Escrito por Luís Miguel Cotrim Mateus - lmmateus@fa.utl.pt *
; *
; *
; * Faculdade de Arquitectura da Universidade Técnica de Lisboa *
; *
; *
; *****
;
; *****
```

### rotação

```
(defun rot (cs sn x y)
  (setq xr (- (* x cs) (* y sn)))
  (setq yr (+ (* x sn) (* y cs)))
)
```

### translação

```
(defun trs (tx ty x y)
  (setq xt (+ x tx))
  (setq yt (+ y ty))
)
```

definição de parâmetros projectivos auxiliares e0, e1, e2, f0, f1, f2, g1, g2  
(defun parametros (Pt1 Pt2 Pt3 Pt4 Pt11 Pt22 Pt33 Pt44)

```
(setq x1 (car Pt1))
(setq y1 (cadr Pt1))
```

```
(setq x2 (car Pt2))
(setq y2 (cadr Pt2))
```

```
(setq x3 (car Pt3))
(setq y3 (cadr Pt3))
```

```
(setq x4 (car Pt4))
(setq y4 (cadr Pt4))
```

```
(setq w1 (car Pt11))
(setq z1 (cadr Pt11))
```

```
(setq w2 (car Pt22))
(setq z2 (cadr Pt22))
```

```
(setq w3 (car Pt33))
(setq z3 (cadr Pt33))
```

```
(setq w4 (car Pt44))
(setq z4 (cadr Pt44))
```

translação dos pontos do 1º quadrilátero com o vector [Pt1 O]

```
(trs (- x1) (- y1) x1 y1)
(setq x1tr xt)
(setq y1tr yt)
```

```
(trs (- x1) (- y1) x2 y2)
(setq x2t xt)
(setq y2t yt)
```

```
(trs (- x1) (- y1) x3 y3)
(setq x3t xt)
(setq y3t yt)
```

```
(trs (- x1) (- y1) x4 y4)
(setq x4t xt)
(setq y4t yt)
```

```
; translação dos pontos do 2º quadrilátero com o vector [Pt44 0]
```

```
(trs (- w4) (- z4) w1 z1)
(setq w1t xt)
(setq z1t yt)
```

```
(trs (- w4) (- z4) w2 z2)
(setq w2t xt)
(setq z2t yt)
```

```
(trs (- w4) (- z4) w3 z3)
(setq w3t xt)
(setq z3t yt)
```

```
(trs (- w4) (- z4) w4 z4)
(setq w4tr xt)
(setq z4tr yt)
```

```
; rotação dos pontos do 1º quadrilátero translaccionados anteriormente,
; com rotação em torno da origem com ângulo definido pelo vector [O P2t]
```

```
(setq dist1 (sqrt (+ (* x2t x2t) (* y2t y2t))))
```

```
(setq cosa1 (/ x2t dist1))
(setq sena1 (/ (- y2t) dist1))
```

```
(rot cosa1 sena1 x2t y2t)
(setq x2tr xr)
(setq y2tr yr)
```

```
(rot cosa1 sena1 x3t y3t)
(setq x3tr xr)
(setq y3tr yr)
```

```
(rot cosa1 sena1 x4t y4t)
(setq x4tr xr)
(setq y4tr yr)
```

```
; rotação dos pontos do 2º quadrilátero translaccionados anteriormente,
; com rotação em torno da origem com ângulo definido pelo vector [O P44t]
```

```
(setq dist2 (sqrt (+ (* w3t w3t) (* z3t z3t))))
```

```
(setq cosa2 (/ w3t dist2))
(setq sena2 (/ (- z3t) dist2))
```

```
(rot cosa2 sena2 w1t z1t)
(setq w1tr xr)
(setq z1tr yr)
```

```
(rot cosa2 sena2 w2t z2t)
(setq w2tr xr)
(setq z2tr yr)
```

```
(rot cosa2 sena2 w3t z3t)
(setq w3tr xr)
(setq z3tr yr)
```

```
; e0, e1, e2, f0, f1, f2, g1, g2
(setq g1 w1tr)
```

```
(setq g2 z1tr)
```

```
(setq e2 (/
  (-
    (* z1tr y3tr)
    (* z1tr y4tr)
  )
  (-
    (* x3tr y4tr)
    (* y3tr x4tr)
  )
)
```

```
(setq f2 (/
  (-
    (* z1tr x4tr)
    (* z1tr x3tr)
  )
  (-
    (* x3tr y4tr)
    (* y3tr x4tr)
  )
)
```

```
(setq e1 (+
  (* e2 (/ w2tr z2tr))
  (* w2tr
    (/
      (- z1tr z2tr)
      (* x2tr z2tr)
    )
  )
  (-
    (/
      (- w1tr w2tr)
      x2tr
    )
  )
)
```

```
(setq f1 (-
  (/
    (+ (* e1 x4tr) w1tr)
    y4tr
  )
)
```

```
(setq e0 (+
  (/ e2 z2tr)
  (/
    (- z1tr z2tr)
    (* x2tr z2tr)
  )
)
```

```
(setq f0 (/
  (+
    (* e1 x3tr)
    (* f1 y3tr)
    w1tr
    (- w3tr)
    (-
      (* e0 x3tr w3tr)
    )
  )
  (* y3tr w3tr)
)
```

#### **:transformação projectiva auxiliar**

```
(defun proj (e0 e1 e2 f0 f1 f2 g1 g2 xntr yntr)
  (setq Xutr (/ (+ (* e1 xntr) (* f1 yntr) g1) (+ (* e0 xntr) (* f0 yntr) 1)))
  (setq Yutr (/ (+ (* e2 xntr) (* f2 yntr) g2) (+ (* e0 xntr) (* f0 yntr) 1)))
)
```

#### **:definição dos pontos de controlo quando se seleccionam linhas**

```
(defun pts_ctrl (la lb)
```

```
(setq n_la (car la))
(setq lla (entget n_la))
(setq IP1la (assoc 10 lla))
(setq IP2la (assoc 11 lla))
(setq e1la (list (cadr IP1la) (caddr IP1la)))
(setq e2la (list (cadr IP2la) (caddr IP2la)))
```

```
(setq n_lb (car lb))
(setq llb (entget n_lb))
(setq IP1lb (assoc 10 llb))
(setq IP2lb (assoc 11 llb))
(setq e1lb (list (cadr IP1lb) (caddr IP1lb)))
(setq e2lb (list (cadr IP2lb) (caddr IP2lb)))
```

```
(setq Pi_a.b (inters e1la e2la e1lb e2lb nil))
)
```

#### **:transformação projectiva**

```
(defun c:transf ()
```

```
(setq -sena2 (- sena2))
```

```
(print "Selecione apenas segmentos de recta...")
```

```
(setq apt (ssget))
(setq npt (sslenght apt))
```

```
(setq kk (- npt 1))
```

```
(while kk
```

```
  (setq nent (ssname apt kk))  
  (setq lent (entget nent))
```

```
  (setq lPn1 (assoc 10 lent))  
  (setq lPn2 (assoc 11 lent))
```

```
  ; 1° endpoint
```

```
  (setq xn1 (cadr lPn1))  
  (setq yn1 (caddr lPn1))
```

```
  (trs (- x1) (- y1) xn1 yn1)  
  (setq xn1t xt)  
  (setq yn1t yt)
```

```
  (rot cosa1 sena1 xn1t yn1t)  
  (setq xn1tr xr)  
  (setq yn1tr yr)
```

```
  (proj e0 e1 e2 f0 f1 f2 g1 g2 xn1tr yn1tr)
```

```
  (setq xu1tr xutr)  
  (setq yu1tr yutr)
```

```
  (rot cosa2 -sena2 xu1tr yu1tr)  
  (setq xu1t xr)  
  (setq yu1t yr)
```

```
  (trs w4 z4 xu1t yu1t)  
  (setq xu1 xt)  
  (setq yu1 yt)
```

```
  (setq lista1 (list 10 xu1 yu1))
```

```
  ; 2° endpoint
```

```
  (setq xn2 (cadr lPn2))  
  (setq yn2 (caddr lPn2))
```

```
  (trs (- x1) (- y1) xn2 yn2)  
  (setq xn2t xt)  
  (setq yn2t yt)
```

```
  (rot cosa1 sena1 xn2t yn2t)  
  (setq xn2tr xr)  
  (setq yn2tr yr)
```

```
  (proj e0 e1 e2 f0 f1 f2 g1 g2 xn2tr yn2tr)
```

```
  (setq xu2tr xutr)  
  (setq yu2tr yutr)
```

```
  (rot cosa2 -sena2 xu2tr yu2tr)  
  (setq xu2t xr)  
  (setq yu2t yr)
```

```
  (trs w4 z4 xu2t yu2t)  
  (setq xu2 xt)  
  (setq yu2 yt)
```

```

(setq lista2 (list 11 xu2 yu2))

(setq lista0 (cons 0 "LINE"))

(setq lista (list lista0 lista1 lista2))

(entmake lista)

(setq kk (- kk 1))

(if (= kk -1) (setq kk nil) )
)
)

```

#### escolha de 4 pontos + 4 pontos e transformação

```

(defun c:4p4p1 ()
  (setq sc-de 1)
  (setq sc-para 0)
  (setq Pt1 (getpoint "\nPonto de Controlo 1:"))
  (setq Pt1 (trans Pt1 sc-de sc-para))
  (setq Pt2 (getpoint "\nPonto de Controlo 2:"))
  (setq Pt2 (trans Pt2 sc-de sc-para))
  (setq Pt3 (getpoint "\nPonto de Controlo 3:"))
  (setq Pt3 (trans Pt3 sc-de sc-para))
  (setq Pt4 (getpoint "\nPonto de Controlo 4:"))
  (setq Pt4 (trans Pt4 sc-de sc-para))
  (setq Pt11 (getpoint "\nPonto correspondente a 1:"))
  (setq Pt11 (trans Pt11 sc-de sc-para))
  (setq Pt22 (getpoint "\nPonto correspondente a 2:"))
  (setq Pt22 (trans Pt22 sc-de sc-para))
  (setq Pt33 (getpoint "\nPonto correspondente a 3:"))
  (setq Pt33 (trans Pt33 sc-de sc-para))
  (setq Pt44 (getpoint "\nPonto correspondente a 4:"))
  (setq Pt44 (trans Pt44 sc-de sc-para))
  (parametros Pt1 Pt2 Pt3 Pt4 Pt11 Pt22 Pt33 Pt44)
  (c:transf)
  (print)
)

```

#### escolha de 1 ponto + 1 ponto (4xs) e transformação

```

(defun c:1p1p4 ()
  (setq sc-de 1)
  (setq sc-para 0)
  (setq Pt1 (getpoint "\nPonto de Controlo 1:"))
  (setq Pt1 (trans Pt1 sc-de sc-para))
  (setq Pt11 (getpoint "\nPonto correspondente a 1:"))
  (setq Pt11 (trans Pt11 sc-de sc-para))
  (setq Pt2 (getpoint "\nPonto de Controlo 2:"))
  (setq Pt2 (trans Pt2 sc-de sc-para))
  (setq Pt22 (getpoint "\nPonto correspondente a 2:"))
  (setq Pt22 (trans Pt22 sc-de sc-para))
  (setq Pt3 (getpoint "\nPonto de Controlo 3:"))
  (setq Pt3 (trans Pt3 sc-de sc-para))
  (setq Pt33 (getpoint "\nPonto correspondente a 3:"))
  (setq Pt33 (trans Pt33 sc-de sc-para))
  (setq Pt4 (getpoint "\nPonto de Controlo 4:"))
  (setq Pt4 (trans Pt4 sc-de sc-para))
  (setq Pt44 (getpoint "\nPonto correspondente a 4:"))
  (setq Pt44 (trans Pt44 sc-de sc-para))
  (parametros Pt1 Pt2 Pt3 Pt4 Pt11 Pt22 Pt33 Pt44)
  (c:transf)
  (print)
)

```

)

#### escolha de 4 linhas + 4 linhas e transformação

```
(defun c:4l4l1 ()  
(print "")  
(setq l1 (entsel "Linha de Controlo 1 não paralela à linha 4:"))  
(print "")  
(setq l2 (entsel "Linha de Controlo 2 não paralela à linha 1:"))  
(print "")  
(setq l3 (entsel "Linha de Controlo 3 não paralela à linha 2:"))  
(print "")  
(setq l4 (entsel "Linha de Controlo 4 não paralela à linha 3:"))  
(print "")  
(setq l11 (entsel "Linha de Controlo correspondente a 1 não paralela à linha correspondente a 4:"))  
(print "")  
(setq l22 (entsel "Linha de Controlo correspondente a 2 não paralela à linha correspondente a 1:"))  
(print "")  
(setq l33 (entsel "Linha de Controlo correspondente a 3 não paralela à linha correspondente a 2:"))  
(print "")  
(setq l44 (entsel "Linha de Controlo correspondente a 4 não paralela à linha correspondente a 3:"))  
  
(pts_ctrl l1 l2)  
(setq Pt1 Pi_a.b)  
(pts_ctrl l2 l3)  
(setq Pt2 Pi_a.b)  
(pts_ctrl l3 l4)  
(setq Pt3 Pi_a.b)  
(pts_ctrl l4 l1)  
(setq Pt4 Pi_a.b)  
  
(pts_ctrl l11 l22)  
(setq Pt11 Pi_a.b)  
(pts_ctrl l22 l33)  
(setq Pt22 Pi_a.b)  
(pts_ctrl l33 l44)  
(setq Pt33 Pi_a.b)  
(pts_ctrl l44 l11)  
(setq Pt44 Pi_a.b)  
  
(parametros Pt1 Pt2 Pt3 Pt4 Pt11 Pt22 Pt33 Pt44)  
(c:transf)  
(print)  
)
```

#### escolha de 1 linha + 1 linha (4xs) e transformação

```
(defun c:1l1l4 ()  
(print "")  
(setq l1 (entsel "Linha de Controlo 1 não paralela à linha 4:"))  
(print "")  
(setq l11 (entsel "Linha de Controlo correspondente a 1 não paralela à linha correspondente a 4:"))  
(print "")  
(setq l2 (entsel "Linha de Controlo 2 não paralela à linha 1:"))  
(print "")  
(setq l22 (entsel "Linha de Controlo correspondente a 2 não paralela à linha correspondente a 1:"))  
(print "")  
(setq l3 (entsel "Linha de Controlo 3 não paralela à linha 2:"))  
(print "")  
(setq l33 (entsel "Linha de Controlo correspondente a 3 não paralela à linha correspondente a 2:"))  
(print "")  
(setq l4 (entsel "Linha de Controlo 4 não paralela à linha 3:"))  
(print "")  
(setq l44 (entsel "Linha de Controlo correspondente a 4 não paralela à linha correspondente a 3:"))
```

```
(pts_ctrl I1 I2)
(setq Pt1 Pi_a.b)
(pts_ctrl I2 I3)
(setq Pt2 Pi_a.b)
(pts_ctrl I3 I4)
(setq Pt3 Pi_a.b)
(pts_ctrl I4 I1)
(setq Pt4 Pi_a.b)
```

```
(pts_ctrl I11 I22)
(setq Pt11 Pi_a.b)
(pts_ctrl I22 I33)
(setq Pt22 Pi_a.b)
(pts_ctrl I33 I44)
(setq Pt33 Pi_a.b)
(pts_ctrl I44 I11)
(setq Pt44 Pi_a.b)
```

```
(parametros Pt1 Pt2 Pt3 Pt4 Pt11 Pt22 Pt33 Pt44)
(c:transf)
(print)
)
```

## Anexo 2 – Código autolisp da rotina para determinação do ângulo entre fachadas

```

*****
;
; *
; * Rotina para determinar o ângulo que fazem duas fachadas *
; * aproximadamente planas através de uma foto de esquina *
; * tirada com o eixo da objectiva na horizontal *
; *
; *****
;
; *****
; *
; * Portugal - Lisboa, Dezembro, 2007 *
; *
; * Escrito por Luis Miguel Cotrim Mateus - lmmateus@fa.utl.pt *
; *
; * Faculdade de Arquitectura da Universidade Técnica de Lisboa *
; *
; *****
;
; distância "dist" de um ponto "Pt1" a uma recta "Pt1.Pt2"
(defun dptr (Pt1 Pt2 Pt3)
  (setq x1 (car Pt1))
  (setq y1 (cadr Pt1))
  (setq x2 (car Pt2))
  (setq y2 (cadr Pt2))
  (setq x3 (car Pt3))
  (setq y3 (cadr Pt3))
  (setq x4 (+ x1 (- y2 y3)))
  (setq y4 (+ y1 (- x3 x2)))
  (setq Pt4 (list x4 y4))
  (setq Pin (inters Pt4 Pt1 Pt2 Pt3 nil))
  (setq dist (distance Pt1 Pin))
)
; intersecção de uma recta (w,z).(w3,0) com uma circunferência de raio R de centro na origem (0,0)
(defun RxC (w z w3 R)
  (setq a (+ 1 (* (/ (- w3 w) z) (/ (- w3 w) z))))
  (setq b (* -2 w3 (/ (- w3 w) z)))
  (setq c (- (* w3 w3) (* R R)))
; ponto 1
  (setq Yu1 (/ (+ (- b) (sqrt (- (* b b) (* 4 a c)))) (* 2 a)))
  (setq Xu1 (- w3 (* Yu1 (/ (- w3 w) z))))
  (setq Pu1 (list Xu1 Yu1))
; ponto 2
  (setq Yu2 (/ (- (- b) (sqrt (- (* b b) (* 4 a c)))) (* 2 a)))
  (setq Xu2 (- w3 (* Yu2 (/ (- w3 w) z))))
  (setq Pu2 (list Xu2 Yu2))
)
; determinante em coordenadas homogéneas de três pontos sendo um deles a origem (x1,y1,1) (x2,y2,1) (0,0,1)
(defun detO (x1 y1 x2 y2)
  (setq dt- (abs (* (+ (* x1 y2) (- (* x2 y1))))))
)
; selecção das medidas
(defun sel (ent)
  (setq i_l (car ent))
  (setq l_e (entget i_l))
  (setq IP1 (assoc 10 l_e))
  (setq IP2 (assoc 11 l_e))
  (setq x1 (cadr IP1))
  (setq y1 (caddr IP1))
  (setq x2 (cadr IP2))
  (setq y2 (caddr IP2))
  (setq P1 (list x1 y1))

```

```

(setq P2 (list x2 y2))
)
; definição dos pontos médios das fachadas na foto
(defun pts_cruz (Phr1 Phr2 P_r1 P_r2)
  (setq Pi_1 (inters Phr1 P_r1 Phr2 P_r2 nil))
  (setq Pi_2 (inters Phr1 P_r2 Phr2 P_r1 nil))

  (setq d1_ (+ (distance Pi_1 Phr1) (distance Pi_1 Phr2) (distance Pi_1 P_r1) (distance Pi_1 P_r2)))
  (setq d2_ (+ (distance Pi_2 Phr1) (distance Pi_2 Phr2) (distance Pi_2 P_r1) (distance Pi_2 P_r2)))

  (if
    (> d1_ d2_)
    (setq Pmr_ Pi_2)
    (setq Pmr_ Pi_1)
  )
)
; define o centro C e raio R das circunferências auxiliares
(defun cen_rai (D_ D_- -d_- -d_-)
  (setq XC (/
    (* -d_- -d_- (- D_ D_-))
    (- (* D_- -d_-) (* D_- -d_-))
  ))
  (setq C (list XC 0))
  (setq R (abs
    (/
      (* D_ D_- (- -d_- -d_-))
      (- (* D_- -d_-) (* D_- -d_-))
    ))
  )
)
; desenho das linhas
(defun linhas (1PI_ 1PL_- 2PI_ 2PL_-)
  (detO (car 1PI_) (cadr 1PI_) (car 1PL_-) (cadr 1PL_-))
  (setq dt11- dt-)

  (detO (car 1PI_) (cadr 1PI_) (car 2PI_-) (cadr 2PI_-))
  (setq dt12- dt-)

  (detO (car 2PI_) (cadr 2PI_) (car 2PL_-) (cadr 2PL_-))
  (setq dt22- dt-)

  (detO (car 2PI_) (cadr 2PI_) (car 1PL_-) (cadr 1PL_-))
  (setq dt21- dt-)

  (setq dt-ref (min dt11- dt12- dt22- dt21-))

  (if (= dt-ref dt11-) (setq Pf1 1PI_)
    (progn
      (if (= dt-ref dt12-) (setq Pf1 1PI_) (setq Pf1 2PI_)))
  )

  (setq XPf1 (car Pf1))
  (setq YPf1 (cadr Pf1))

  (setq XPf1- (+ XPf1 Bx))
  (setq YPf1- (+ YPf1 By))

  (setq lista2 (list 11 XPf1- YPf1-))
  (setq lista (list lista0 lista1 lista2))
  (entmake lista)
)

```

```

)
;rotina para desenhar o angulo entre duas fachadas verticais com uma aresta comum
(defun c:planta ()
;      distancias de controlo
(print "")
(setq ent (entsel "Na foto, seleccione a aresta comum às duas fachadas:"))
(sel ent)
(setq Phr1 P1)
(setq Phr2 P2)
(setq hr (distance Phr1 Phr2))
(print "")
(setq ent (entsel "Na foto, seleccione uma vertical à esquerda da aresta comum:"))
(sel ent)
(setq Per1 P1)
(setq Per2 P2)
(print "")
(setq ent (entsel "Na foto, seleccione uma vertical à direita da aresta comum:"))
(sel ent)
(setq Pdr1 P1)
(setq Pdr2 P2)
(print "")
(setq ent (entsel "Seleccione num dos alçados a altura de referência:"))
(sel ent)
(setq Ph1 P1)
(setq Ph2 P2)
(setq h (distance Ph1 Ph2))
(print "")
(setq ent (entsel "Seleccione a largura da fachada esquerda:"))
(sel ent)
(setq Pe1 P1)
(setq Pe2 P2)
(setq D1 (distance Pe1 Pe2))
(print "")
(setq ent (entsel "Seleccione a largura da fachada direita:"))
(sel ent)
(setq Pd1 P1)
(setq Pd2 P2)
(setq D2 (distance Pd1 Pd2))
(print "")

(dptr Per1 Phr1 Phr2)
(setq -d1r dist)

(dptr Pdr1 Phr1 Phr2)
(setq -d2r dist)

(setq D1- (/ D1 2))
(setq D2- (/ D2 2))

(setq -d1 (* -d1r (/ h hr)))
(setq -d2 (* -d2r (/ h hr)))
;      pontos de cruzamento na fachada esquerda, foto
(pts_cruz Phr1 Phr2 Per1 Per2)
(setq Pmre Pmr_)
(dptr Pmre Phr1 Phr2)
(setq -d1- (* dist (/ h hr)))
;      pontos de cruzamento na fachada direita, foto
(pts_cruz Phr1 Phr2 Pdr1 Pdr2)
(setq Pmrd Pmr_)
(dptr Pmrd Phr1 Phr2)
(setq -d2- (* dist (/ h hr)))
;      vértice da planta

```

```

(setq BB (getpoint "Invértice da planta:"))
  (setq sc-de 1)
  (setq sc-para 0)
    (setq BB (trans BB sc-de sc-para))
  (setq Bx (car BB))
  (setq By (cadr BB))
; listas para a criação da entidade linha
  (setq lista0 (cons 0 "LINE"))
  (setq lista1 (list 10 Bx By))
; centro C1 da circunferência 1 de raio R1
(cen_rai D1 D1- -d1 -d1-)
  (setq C1 (list (- (car C)) 0))
  (setq R1 R)
; centro C2 da circunferência 2 de raio R2
(cen_rai D2 D2- -d2 -d2-)
  (setq C2 C)
  (setq R2 R)
; determinação dos pontos I1 e I2 de intersecção das circunferências de raio R1 e R2, com centros em
C1 e C2, respectivamente
; operação equivalente à determinação da posição do Observador; na prática só é necessário I2
(setq Xal (- (car C1) R1))
(setq Xal+ (+ (car C1) R1))
(setq Xbe (- (car C2) R2))
(setq Xbe+ (+ (car C2) R2))

(setq Xi (/ (- (* Xal Xal-) (* Xbe Xbe-)) (+ Xal- Xal (- Xbe-) (- Xbe))))
(setq Yi1 (sqrt (* (- Xal- Xi) (- Xi Xal))))
(setq Yi2 (- Yi1))

(setq I2 (list Xi Yi2))
; determinação das posições possíveis para os pontos da planta (lado esquerdo)
; sobre as linhas [I2 P1 ] e [I2 P1-]
(setq x1 (- -d1))
(RxC Xi Yi2 x1 D1)
  (setq 1PI1 Pu1)
  (setq 2PI1 Pu2)
(setq x1- (- -d1-))
(RxC Xi Yi2 x1- D1-)
  (setq 1PI1- Pu1)
  (setq 2PI1- Pu2)
; determinação da linha do lado esquerdo
(linhas 1PI1 1PI1- 2PI1 2PI1-)
; determinação das posições possíveis para os pontos da planta (lado direito)
; sobre as linhas [I2 P2 ] e [I2 P2-]
(setq x2 -d2)
(RxC Xi Yi2 x2 D2)
  (setq 1PI2 Pu1)
  (setq 2PI2 Pu2)
(setq x2- -d2-)
(RxC Xi Yi2 x2- D2-)
  (setq 1PI2- Pu1)
  (setq 2PI2- Pu2)
; determinação da linha do lado direito
(linhas 1PI2 1PI2- 2PI2 2PI2-)
;fim
(print)
)

```

