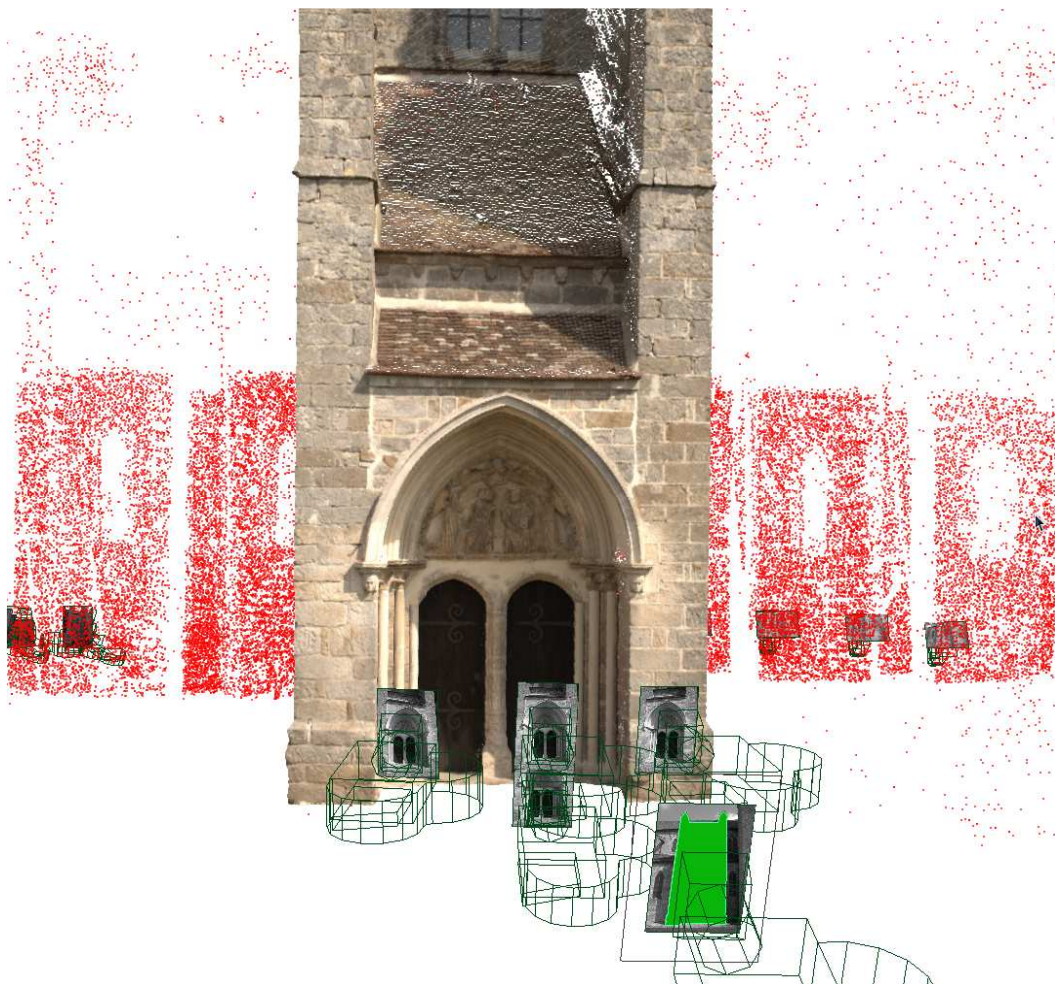


PARAMETRAGE DE LA CHAÎNE PAM



Aymeric GODET

SOMMAIRE

Ce document est la synthèse de la semaine de formation à l'IGN donnée par monsieur Marc Pierrot Deseilligny sur les fichiers de paramètres de la chaîne PAM. Il a été le fondement à partir duquel PhotoCloud rédige automatiquement les fichiers XML de paramètre.

1. Organisation et prétraitements des données du chantier.....	3
2. Paramétrage des données :.....	4
3. PASTIS.....	5
a) Présentation de la commande :.....	5
b) Utilisation dans la chaîne :.....	6
4. APERO	6
a) Présentation de la commande.....	6
b) Fichier de paramétrage	7
5. MICMAC	11
a) Présentation de la commande.....	11
b) Fichier de paramétrage de MICMAC.....	12
c) Fichier xml de calibration.....	4

1. Organisation et prétraitements des données du chantier

Les images acquises par la caméra sont au format RAW avec des informations sur la prise de vue dans l'EXIV associé. Le prétraitement à appliquer ici est l'ajout d'un préfixe contenant la focale utilisée. Cette opération est effectuée avec la commande :

```
MyRename "/CheminDuChantier/(.*)RAW" "F\$2_\$1.RAW" AddFoc=1 Exe=1
```

Remarque:

Toutes les lignes de commande doivent être faites dans le dossier Chantier.

RAW : extension du fichier RAW (CR2, NEF ...)

Le second traitement à appliquer pour être compatible avec PASTIS est de générer les images TIF à partir des RAW. Nous allons donc débayeriser les images avec la commande :

```
MpDcraw IMAGE.RAW 16B=0 GB=1
```

Remarque :

16B : encoder ou non en 16 bit

GB=1 : Niveau de gris (ou CB=1 -> Image en couleur)

Cette commande ne travaille que sur une image. Nous pouvons utiliser un MapCmd afin de traiter tout le chantier d'un coup:

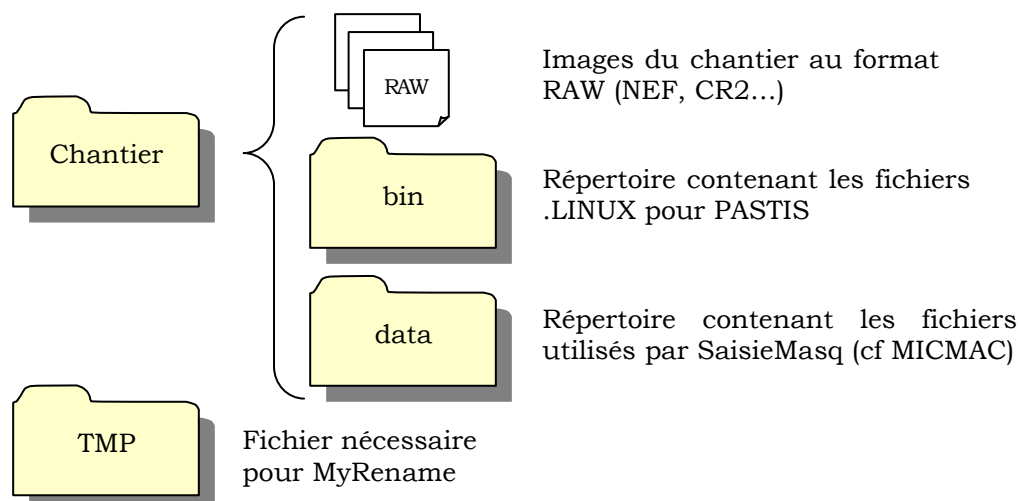
```
MapCmd MpDcraw "P=(.*)RAW" 16B=0 GB=1 M=Mk "T=\$1_MpDcraw8B_GB.tif"
```

Un fichier makefile Mk est crée et peut être lancé par

```
make all -f Mk -j(NBPROC)
```

où (NBPROC) est le nombre de processeur afin de travailler en parallèle.

Schéma d'organisation :



2. Paramétrage des données :

La philosophie du paramétrage de la chaîne de traitement de PASTIS APERO et MICMAC se fonde sur l'utilisation de fichiers XML. Les paramètres sont contenus au sein de balises et un vocabulaire peut ainsi être développé : ce sont des Clés. La description exhaustive des options de paramétrage d'APERO et de MICMAC sont contenues dans le dossier include\XML_GEN du dossier d'installation. Nous présentons ici les paramètres les plus pertinents à notre contexte.

Un exemple:

Le fichier MicMac-LocalChantierDescripteur.xml présent dans le dossier Chantier définit la plupart des clés utilisées pour le traitement. La clé décrite ici sert à définir l'ensemble des images du chantier de focale 20mm en utilisant des patronnes (cf annexes) :

```
<KeyedSetsOfNames>
  <Sets>
    <PatternAccepteur>
      F020_DSC_[0-9]{4}_MpDcraw8B_GB.tif
    </PatternAccepteur>
  </Sets>
</Key>
  Key20Img
</Key>
</KeyedSetsOfNames>
```

Il existe un dictionnaire de clé par défaut présent dans le dossier d'installation, DefautChantierDescripteur.xml, qui peut être utilisé.

Fichier xml de calibration

Les fichiers de calibration utilisés par APERO et MICMAC reprennent les différentes notions que nous venons d'aborder. Voici un exemple de calibration :

```
<ExportAPERO>
  <CalibrationInternConique>
    <KnownConv>eConvApero_DistM2C</KnownConv>
    <PP>2798.64782210431758 1874.713396725145</PP>
    <F>5584.25100584813299</F>
    <SzIm>5634 3753</SzIm>
    <CalibDistortion>
      <ModRad>
        <CDist>2784.85598883226339 1873.95782987736129</CDist>
        <CoeffDist>-2.76667760178986444e-09</CoeffDist>
        <CoeffDist>1.08484268610007447e-16</CoeffDist>
        <CoeffDist>5.61875301076978648e-25</CoeffDist>
      </ModRad>
    </CalibDistortion>
  </CalibrationInternConique>
</ExportAPERO>
```

<PP> : Ce sont les coordonnées du PPA en pixel

<F> : C'est la valeur de la distance principale en pixel

<SzIm> : Taille de l'image

<CDist> : Ce sont les coordonnées du PPS en pixel

<CoeffDist> : Ce sont les coefficients (a,b,c) du modèle de distorsion

Lors du traitement nous devons initialiser les calibrations. Nous approximerons la calibration en paramétrant de la façon suivante :

- Les points principaux sont assimilés au centre de l'image
- Il n'y a pas de distorsion (ie $a=b=c=0$)

3. PASTIS

a) Présentation de la commande :

PASTIS est l'acronyme de **P**rogramme utilisant **A**utopano **S**ift pour les **T**ie-points dans les **I**mage**S**. C'est lors de cette étape que nous allons détecter les points SIFT dans un groupe d'images défini par une clé. La commande de Pastis s'effectue en deux temps :

```
Pastis /CheminDuChantier/ CléDeGroupeImage TailleSousEchantillon MkF=Mk Opt
```

CléDeGroupeImage : clé définie dans le dictionnaire (cf exemple)

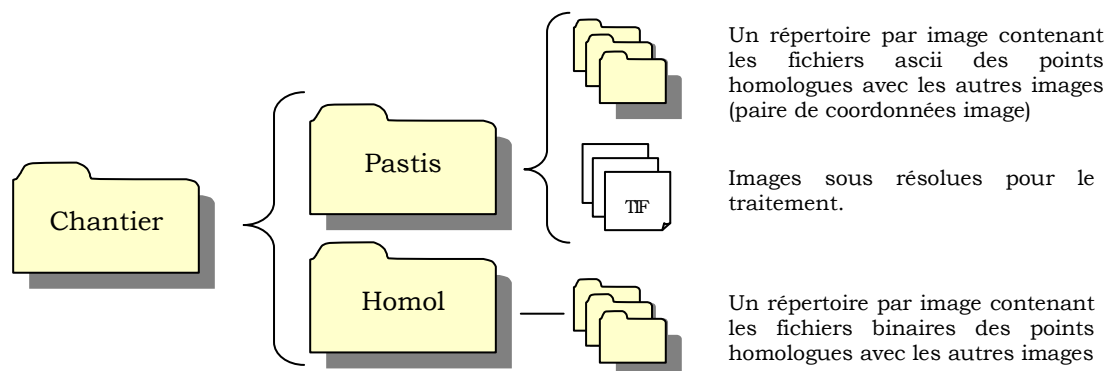
TailleSousEchantillon : Entier représentant la largeur des images sur lesquelles PASTIS va travailler.

MkF : Nom du fichier Makefile générer en fin de programme

Opt : Options supplémentaires pouvant être ajouté.

Lors de cette passe le logiciel va créer toute l'arborescence nécessaire au traitement dans le dossier Chantier ainsi que les images sous résolues.

L'arborescence est de la forme :



Après avoir lancé le makefile en parallèle pour optimiser le temps de calcul, chaque sous répertoire va contenir les différents fichiers de points homologues.

b) Utilisation dans la chaîne :

La stratégie utilisée dans la chaîne de traitement pour optimiser le temps de calcul peut s'appuyer sur plusieurs paramètres :

- Un a priori sur les images voisines :

Au lieu de tester toutes les possibilités de combinaison du chantier on pourrait par exemple définir des clés spécifiques de groupe d'images par site de corrélation avec seulement les images de prise de vue global potentiellement concernées.

- La taille du sous échantillonnage :

Il peut être intéressant de travailler avec des images sous résolues afin de limiter le temps des calculs. Il ne faut cependant pas descendre trop bas en qualité de peur de perdre trop d'informations. Nous avons par exemple traité les chantiers en deux passes en augmentant la résolution la seconde fois tout en filtrant les couples avec les résultats de la première passe.

- En utilisant les options de PASTIS :

SsRes

Booléen permettant de préciser que l'on travaille en sous-résolution afin d'optimiser les algorithmes mis en œuvre.

NbMinPtExp=X

Un fichier de points homologues est créé si au moins X points homologues sont détectés.

4. APERO

a) Présentation de la commande

APERO est l'acronyme de **A**érotriangulation **P**hotogrammétrique **E**xpérimentale **R**elativement **O**pérationnelle. Ce logiciel sert à calculer l'aérotriangulation d'un chantier (détermination des positions et orientations des caméras). La commande pour lancer le traitement est :

```
Apero fichierDeParametre.xml
```

Toute la description du chantier est contenue dans le fichier xml de paramétrage présenté en argument dans la commande.

b) Fichier de paramétrage

Le fichier de paramétrage doit être situé dans le dossier Chantier. Nous allons présenter ici le détail des différents nœuds du fichier.

```
<ParamApero>
  <SectionBDD_Observation>
  <SectionInconnues>
  <SectionChantier>
  <SectionSolveur>
  <SectionCompensation>
</ParamApero>
```

Les quatre premières sections posent le problème du chantier et la dernière présente la manière dont les calculs et les compensations vont être réalisés.

i. SectionBDD_Observation

Dans cette section sont définis les points d'appuis (par topométrie...) et de liaisons (PASTIS) du chantier nécessaires au calcul. S'il n'y a aucun point d'appui l'aérotriangulation sera faite en l'air.

Un exemple de paramétrage :

```
<BDD_PtsLiaisons>
  <Id> Id_Pastis_Hom </Id>
  <KeySet> Key-Set-HomolPastisBin </KeySet>
  <KeyAssoc> Key-Assoc-Cplelm2HomolPastisBin </KeyAssoc>
  <UseAsPtMultiple> true </UseAsPtMultiple>
</BDD_PtsLiaisons>
```

<Id> : Nom sous lequel sera référencé cet ensemble de données
<KeySet> : Clé désignant les dossiers où sont les données. Dans cet exemple elle désigne les dossiers PASTIS et HOMOL. Cette clé est définie dans le dictionnaire par défaut.

<KeyAssoc> : Clé du dictionnaire par défaut spécifiant comment sont organisés les fichiers de points de liaison pour un couple de photos.

<UseAsPtMultiple> : Booléen pour utiliser les points comme points multiples (points vues sur plus de deux images).

ii. SectionInconnues

Dans cette section sont définies les inconnues du système à savoir la calibration, les positions et orientations des caméras.

Exemple de calibration :

```
<CalibrationCameraInc>
  <Name> TheKeyCalib_035 </Name>
  <CalValueInit>
    <CalFromFileExtern>
      <NameFile> Auto-Calib-035.xml </NameFile>
      <NameTag> CalibrationInternConique </NameTag>
    </CalFromFileExtern>
  </CalValueInit>
</CalibrationCameraInc>
```

<Name> : Nom sous lequel sera référencée cette caméra

<CalFromFileExtern> : Fichier des paramètres initiaux de la caméra.

<NameTag> : modèle géométrique de la caméra

Exemple de position de caméra de référence:

Nous définissons ici arbitrairement une prise de vue qui sera fixée afin de pouvoir réaliser une aérotriangulation en l'air.

```
<PoseCameraInc>
  <PatternName> F035_img_0916_MpDcraw8B_GB.tif </PatternName>
  <CalcNameCalib> TheKeyCalib_035 </CalcNameCalib>
  <PosValueInit>
    <PosId> ### </PosId>
  </PosValueInit>
</PoseCameraInc>
```

<PatternName> : Nom du fichier image associé à cette configuration

<CalcNameCalib> : Nom de la calibration associée à l'image.

<PosId> : Cette image aura la position identité.

Exemple d'un ensemble d'images :

```
<PoseCameraInc>
  <PatternName> F035_img_0[8-9][0-9]{2}_MpDcraw8B_GB.tif </PatternName>
  <AutoRefutDupl> true </AutoRefutDupl>
  <CalcNameCalib> TheKeyCalib_035 </CalcNameCalib>
  <PosesDeRattachement> 0 </PosesDeRattachement>
  <InitNow> false </InitNow>
  <MEP_SPEC_MST>
    <Show> true </Show>
  </MEP_SPEC_MST>
  <PosValueInit>
    <PoseFromLiaisons>
      <LiaisonsInit>
        <NameCam> ### </NameCam>
        <IdBD> Id_Pastis_Hom </IdBD>
      </LiaisonsInit>
    </PoseFromLiaisons>
  </PosValueInit>
</PoseCameraInc>
```

<PatternName> : Le groupe d'images est défini à l'aide de paterne.

<AutoRefutDupl> : Afin de pouvoir gérer les doublons

<PosesDeRattachement> : Permet de gérer les contraintes de bases arbitraires.

<InitNow> : l'initialisation ne se fait pas tout de suite afin de pouvoir compenser

<MEP_SPEC_MST> : Nous spécifions ici l'algorithme de choix automatique de mise en place.

<LiaisonsInit> : Nous spécifions ici quelle base de donnée de point est à utiliser pour ce groupe d'image.

iii. SectionChantier

Dans cette section nous précisons où est situé le fichier.

Exemple :

```
<SectionChantier>
  <DirectoryChantier> ThisDir </DirectoryChantier>
</SectionChantier>
```

La clé ThisDir fait référence au dossier où est placé le fichier de paramètre

iv. SectionSolveur

Dans cette section nous spécifions quelle stratégie de résolution nous utilisons.

Exemple :

```
<SectionSolveur>
  <ModeResolution> eSysPlein </ModeResolution>
</SectionSolveur>
```

v. SectionCompensation

Cette section présente l'organisation de la compensation que va réaliser APERO. Elle contient différentes étapes de compensations et suit une ossature du type :

```
<EtapeCompensation>
  <SectionObservations>
  <IterationsCompensation>
  ...
  <IterationsCompensation>
  <SectionExport>
</EtapeCompensation>
```

Nous allons dans la suite du document détailler chacune de ces branches.

vi. Section des observations

Dans cette section sont renseignés l'identifiant de la base de données et les paramètres statistiques de la compensation.

Exemple :

```
<ObsLiaisons>
  <NameRef> Id_Pastis_Hom </NameRef>
  <Pond>
    <EcartMesureIndiv> 1.0 </EcartMesureIndiv>
    <Show> eNSM_Paquet </Show>
    <NbMax> 100 </NbMax>
    <EcartMax> 30 </EcartMax>
    <SigmaPond> 5 </SigmaPond>
    <ModePonderation> eL1Secured </ModePonderation>
  </Pond>
</ObsLiaisons>
```

<NameRef> : Nom de la base de données utilisée
<EcartMesureIndiv> : Poids global utilisé
<NbMax> : Poids équivalent à partir de X liaisons. Ce bornage est réalisé afin de placer chaque image ayant un nombre suffisant de liaison au même degré de pondération.

<EcartMax> : Les points dont les résidus sont supérieurs à X sont écartés du calcul.

<SigmaPond> : C'est l'écart type utilisé dans la pondération.

<ModePonderation> C'est le mode de pondération utilisé.

(a) Section des itérations

Pour chaque nouvelle balise <IterationsCompensation> une nouvelle compensation est réalisée. La première itération est un peu particulière car elle va spécifier toutes les contraintes initiales.

Exemple d'itération initiale :

```
<IterationsCompensation>
  <Pose2Init>
    <ProfMin> [2,4,6,8,10,12,14,16,18,20,22,24,26,30,32,34,36] </ProfMin>
    <Show> true </Show>
  </Pose2Init>
  <SectionContraintes>
    <ContraintesCamerasInc>
      <Val> eLiberteFocale_0 </Val>
      <Val> eLib_PP_CD_00 </Val>
      <Val> eLiberte_DR0 </Val>
    </ContraintesCamerasInc>
    <ContraintesPoses>
      <NamePose> 0 </NamePose>
      <Val> ePoseFigee </Val>
    </ContraintesPoses>
    <ContraintesPoses>
      <NamePose> 1 </NamePose>
      <Val> ePoseBaseNormee </Val>
      <PoseRattachement> 0 </PoseRattachement>
    </ContraintesPoses>
  </SectionContraintes>
</IterationsCompensation>
```

<ProfMin> : Dans cette liste sont contenues les différentes profondeurs d'image que vont parcourir les itérations initiales. Pour chaque nouvelle profondeur les images inférieures ou égales à cette profondeur seront ajoutées au système. La profondeur d'une image est fonction de celle de ses images parents qui est une mesure de distance (au sens large) par rapport à une image de référence de profondeur nulle.

<ContraintesCamerasInc> : Ici nous définissons les contraintes associées à la calibration des caméras du chantier. Dans l'exemple présenté la focale, le PPA, le PPS et les paramètres de distorsion sont fixés à la calibration initiale.

<ContraintesPoses> : La balise <NamePose> est un index faisant référence aux groupes d'images définis dans la section des inconnues. 0 correspond dans notre exemple à l'image de référence qui devient fixée par la balise <Val>ePoseFigee</Val>.

Exemple d'itération au milieu de l'étape de compensation :

```
<IterationsCompensation>
  <SectionContraintes>
    <ContraintesCamerasInc>
      <Val> eLiberte_DR1 </Val>
    </ContraintesCamerasInc>
  </SectionContraintes>
</IterationsCompensation>
```

A partir de cette itération le premier paramètre de distorsion est libéré.

Remarque :

<IterationsCompensation></IterationsCompensation>

Dans ce cas l'itération est effectuée avec les mêmes contraintes que dans le calcul précédent.

(b) Section d'export

Cette section est réservée aux informations du calcul qu'il est possible d'exporter. Nous pouvons exporter la calibration, la position et orientation des caméras.

Exemple d'export :

```
<SectionExport>
  <ExportPose>
    <PatternSel> (.*)tif </PatternSel>
    <KeyAssoc> Key-Assoc-Im2OrInit </KeyAssoc>
    <AddCalib> true </AddCalib>
    <NbVerif> 10 </NbVerif>
    <TolWhenVerif> 1e-3 </TolWhenVerif>
    <FileExtern> Key-Assoc-Im2AutoCallnit </FileExtern>
    <FileExternIsKey> true </FileExternIsKey>
  </ExportPose>

  <ExportCalib>
    <KeyAssoc> Key-Assoc-CleCal2AutoCallnit </KeyAssoc>
    <KeyIsName> false </KeyIsName>
  </ExportCalib>
</SectionExport>
```

<ExportPose> : Export des positions et orientations des images

<ExportCalib> : Export de la calibration

5. MICMAC

a) Présentation de la commande

MICMAC est l'acronyme de **M**ulti-**I**mages **C**orrespondances, **M**éthodes **A**utomatiques de **C**orrélation. C'est avec cet outil que nous allons réaliser les nuages de points des sites de corrélations du chantier. Tout comme APERO la commande de lancement prend en argument un fichier xml de paramétrage :

MICMAC fichierDeParmètre.xml

Le seul fichier de paramétrage est en réalité insuffisant afin de traiter un site de corrélation. En effet nous devons joindre au traitement un masque de l'image maitresse. Ce masque est une image TIF binaire associée à un fichier xml correspondant aux zones de l'image maitresse sur lesquelles MICMAC va corréler. A l'heure actuelle nous utilisons le logiciel SaisieMasq pour réaliser de tels masques mais nous devons certainement moderniser ce processus.

b) Fichier de paramétrage de MICMAC

Il existe un langage dans le fichier de paramétrage de MICMAC qui permet de gérer des variables afin de traiter en série et/ou en parallèle une suite de site de corrélation. Ces variables sont définies par défaut dans la balise de ParamMICMAC (cf les balises ci-dessous) et redéfinies au cas par cas dans la section Workspace. Elles seront reconnaissables dans les exemples à suivre car elles sont encapsulées par un $\{\dots\}$.

Nous allons dans la suite du document vous présenter les différentes branches contenues dans le fichier de paramétrage :

```
<ParamMICMAC>
  <Section_Terrain>
  <Section_PriseDeVue >
  <Section_MEC >
  <Section_Results >
  <Section_WorkSpace >
</ParamMICMAC>
```

i. Variable par défaut

C'est au sein de la balise de ParamMICMAC que nous allons initialiser les variables utilisées dans l'optique d'un traitement en série.

Un exemple :

```
<ParamMICMAC
  Subst="@ $#1"
  NameDecl="@ $#1"
  NumImC="@XXXXX"
  NumImV="@XXXX"
  ScaleNuage="@4"
  Foc="@035">
```

ii. Section_Terrain

Dans cette section nous nous intéressons à définir l'espace de recherche de MICMAC avec des contraintes sur les parallaxes attendues et le masque de l'image maitresse.

Un exemple :

```
<Section_Terrain>
  <IntervSpecialZInv >
    <MulZMin> 0.5 </MulZMin>
    <MulZMax> 3 </MulZMax>
  </IntervSpecialZInv>
  <Planimetrie>
    <MasqueTerrain >
      <MT_Image> F$\{Foc\}_img_{\{NumImC\}}_MpDcraw8B_GB_Masq.tif </MT_Image>
      <MT_Xml> F$\{Foc\}_img_{\{NumImC\}}_MpDcraw8B_GB_Masq.xml </MT_Xml>
    </MasqueTerrain>
  </Planimetrie>
</Section_Terrain>
```

<IntervSpecialZInv > : Nous réglons ici l'intervalle de parallaxe en relatif par rapport à la profondeur moyenne sur lequel MICMAC va travailler.

<MasqueTerrain > : Ici nous spécifions l'image masque à appliquer à l'image maitresse. Vous pouvez constater ici une première utilisation des variables.

iii. Section_PriseDeVue

Dans cette section nous définissons les images du site de corrélation à traiter.

Un exemple :

```
<Section_PriseDeVue >
  <GeomImages> eGeomImageOri </GeomImages>
  <Images>
    <Im1 > F${Foc}_img_${NumImC}_MpDcraw8B_GB.tif </Im1>
    <ImPat > F${Foc}_img_${NumImV}_MpDcraw8B_GB.tif </ImPat>
  </Images>
  <NomsGeometriImage>
    <FCND_Mode_GeomIm>
      <FCND_GeomCalc> Key-Assoc-Im2Orlnit </FCND_GeomCalc>
    </FCND_Mode_GeomIm>
  </NomsGeometriImage>
</Section_PriseDeVue>
```

<GeomImages> : nous spécifions la géométrie de l'image (ici conique).

<Im1 > : Lien vers l'image maitresse.

<ImPat > : Lien vers les autres images du site actuel de corrélation.

<FCND_GeomCalc> Ici nous avons une clé en référence à la calibration des images.

iv. Section_MEC

La section MEC, **Mise En Correspondance**, présente la stratégie de corrélation que va suivre MICMAC. Il suit dans la structure le même état d'esprit que la section de compensation d'APERO. Nous avons une étape d'initialisation suivit de plusieurs passes de traitement sur une pyramide d'image, ce sont des étapes de mise en correspondance.

Ainsi une section de mise en correspondance est de la forme :

```
<Section_MEC >
  <ChantierFullImage1>
  <TypePyramImage>
  <EtapeMEC>
  ...
  <EtapeMEC>
</Section_MEC >
```

La première balise requiert un booléen pour savoir si l'image de niveau 1 est à pleine résolution. La seconde définit la pyramide d'image avec le saut de résolution par niveau et le nombre de bit d'encodage des images.

(a) Initialisation :

L'initialisation s'effectue dans la première Etape de MEC.

Exemple :

```
<EtapeMEC>
  <DeZoom > -1          </DeZoom>
  <DynamiqueCorrel> eCoeffCorrelStd </DynamiqueCorrel>
  <AggregCorr> eAggregSymetrique </AggregCorr>
  <SzW > 2              </SzW>
  <AlgoRegul> eAlgo2PrgDyn </AlgoRegul>
  <ZRegul> 0.05         </ZRegul>
  <ZPas> 0.5            </ZPas>
  <ZDilatAlti> 3         </ZDilatAlti>
  <ZDilatPlani> 4        </ZDilatPlani>
  <ModulationProgDyn>
    <EtapeProgDyn>
      <ModeAgreg> ePrgDAgrSomme </ModeAgreg>
      <NbDir> 6          </NbDir>
    </EtapeProgDyn>
    <Px1PenteMax> 10.0   </Px1PenteMax>
  </ModulationProgDyn>
  <GenImagesCorrel> true </GenImagesCorrel>
</EtapeMEC>
```

<DeZoom > : Le niveau actuel dans la pyramide d'image. La valeur -1 est réservée à l'initialisation.

<DynamiqueCorrel> : La façon dont le coefficient de corrélation est calculé.

<AggregCorr> : Dans notre exemple toutes les images ont un rôle symétrique dans le calcul.

<SzW > : Taille de la demie fenêtre de corrélation moins un.

<AlgoRegul> : algorithme de régularisation utilisé

<ZRegul> : échelle de la régularisation

<ZPas> : Pas de quantification pour la recherche

<ZDilatAlti> : Zone de recherche en profondeur pour la nouvelle passe en fonction de la solution de la précédente passe.

<ZDilatPlani> : Zone de recherche dans le plan pour la nouvelle passe en fonction de la solution de la précédente passe.

<NbDir> Nombre de direction dans lequel MICMAC va chercher des correspondances.

<GenImagesCorrel> Booléen pour la génération des images du score de corrélation.

(b) Etape de mise en correspondance :

La plupart des paramètres ayant été définis dans la phase d'initialisation on ne spécifie généralement que le niveau de la pyramide d'image à traiter.

v. Section_Results

Dans cette section nous précisons quels résultats nous souhaitons obtenir.

Exemple :

```
<Section_Results >
  <GeomMNT> eGeomMNTFaisceaulm1PrCh_Px1D </GeomMNT>
  <DoMEC > true </DoMEC>
  <ZoomMakeTA> 8 </ZoomMakeTA>
  <GammaVisu> 2.0 </GammaVisu>
  <ZoomVisuLiaison> -1 </ZoomVisuLiaison>
</Section_Results>
```

<GeomMNT> : Nous spécifions dans quelle géométrie nos résultats sont obtenus. En l'occurrence nous nous plaçons dans la géométrie image.

vi. Section_WorkSpace

Dans la section WorkSpace nous organisons l'archivage et le déroulement des sites à accomplir. Nous spécifions également quelques informations hardware afin d'optimiser le traitement. Enfin dans le dernier nœud de cette section nous implémentons les post-traitements à appliquer notamment pour l'export.

Un exemple

```
<Section_WorkSpace >
```

Dans cette partie nous allons spécifier les répertoires de sortie.

```
<TmpMEC > Geol-${NumImC}/ </TmpMEC>
<TmpPyr > Geol-${NumImC}/ </TmpPyr>
<TmpResult > Geol-${NumImC}/ </TmpResult>
<WorkDir > ThisDir </WorkDir>
```

Ici nous spécifions le nombre de processeur à utiliser et la mémoire vive utilisable.

```
<ByProcess> 6 </ByProcess>
<AvalaibleMemory> 1024 </AvalaibleMemory>
```

MICMAC découpe les images à traiter en dalle, nous spécifions ici ces caractéristiques.

```
<SzDalleMin> 400 </SzDalleMin>
<SzDalleMax> 1000 </SzDalleMax>
<SzRecouvrtDalles> 100 </SzRecouvrtDalles>
<FirstEtapeMEC> 0 </FirstEtapeMEC>
```

Dans ce nœud nous définissons les différents sites de corrélation à calculer. Les variables utilisées sont définies à chaque nouvelle passe dans la balise <CMVA>.

```
<MapMicMac>
  <ActivateCmdMap> true </ActivateCmdMap>
  <ModeCmdMapeur>
    <CM_One> toto </CM_One>
  </ModeCmdMapeur>
  <CMVA>
    <NV> Foc "035" </NV><NV> NumImC "0953" </NV><NV> NumImV "095[4-6]" </NV>
  </CMVA>
  <CMVA>
    <NV> Foc "035" </NV><NV> NumImC "0957" </NV><NV> NumImV "09(58|59|60)" </NV>
  </CMVA>
</MapMicMac>
```

Cette dernière partie correspond au post traitement. Dans cet exemple nous réalisons une image couleur de l'image maitresse afin de pouvoir colorier le nuage de point que nous exportons au format PLY. La dernière commande déplace et renomme le nuage pour une meilleure gestion de l'archivage.

```
<PostProcess>
  <OneCmdPar>
    <OneCmdSer>
      bin/MpDcraw ${ThisDir}F${Foc}_IMG_${NumImC}.CR2 16B=0 CB=1
    </OneCmdSer>
    <OneCmdSer>
      bin/Nuage2Ply ${ThisDir}Geol-
        ${NumImC}/NuageImProf_LeChantier_Etape_8.xml Scale=${ScaleNuage}
        Attr=${ThisDir}F${Foc}_IMG_${NumImC}_MpDcraw8B_CB.tif
    </OneCmdSer>
    <OneCmdSer>
      mv ${ThisDir}Geol-${NumImC}/NuageImProf_LeChantier_Etape_8.ply
        ${ThisDir}PLY/Nuage-${NumImC}.ply </OneCmdSer>
    </OneCmdPar>
  </PostProcess>
</Section_WorkSpace >
```