```
(*geometric way for Triangulation_____________________________________*)
(*_____________________________________________________________________*)

CreateTriangulation[IPC1_, IPC2_, P22_] :=
  Module[{LinesC1, LinesC2, PMW, O22D, O2D, V, ImagePlaneC2PointsWorld,
   Modul
    ImagePlaneC1PointsWorld, solve, ReconstructedPointsC1, ReconstructedPointsC2,
    C2Oc2, WOc2, test, ResizedPointsC1, ResizedPointsC2, scaleValueC1,
    scaleValueC2, Ro2, GraphicPointsC1, GraphicPointsC2, G1, G2},
   Print["Triangulation: WorldPoint reconstruction
   gib aus
      _____________________________________________________"];

   O2D = {Oc[[1]], Oc[[2]], Oc[[3]]};
   C2Oc2 = {P22[[1, 4]], P22[[2, 4]], P22[[3, 4]]};

   Print["C2Oc2 = ", C2Oc2];
   gib aus

   If[beta ≠ 0,
   wenn
    Ro2 = {{P22[[1, 1]], P22[[1, 2]], P22[[1, 3]]}, {P22[[2, 1]],
       P22[[2, 2]], P22[[2, 3]]}, {P22[[3, 1]], P22[[3, 2]], P22[[3, 3]]}};
    Ro2 = Transpose[Ro2];
          transponiere
    WOc2 = -Ro2.C2Oc2;
    Print["Ro2 = ", MatrixForm[Ro2]];
    gib aus              Matritzenform
    Print["WOc2 = ", WOc2];
    gib aus
    PMW = {{Ro2[[1, 1]], Ro2[[1, 2]], Ro2[[1, 3]], WOc2[[1]]},
      {Ro2[[2, 1]], Ro2[[2, 2]], Ro2[[2, 3]], WOc2[[2]]},
      {Ro2[[3, 1]], Ro2[[3, 2]], Ro2[[3, 3]], WOc2[[3]]}, {0, 0, 0, 1}};
   ];

   If[beta == 0,
   wenn
    WOc2 = -Rot1.C2Oc2;
    Print["-Rot1 = ", MatrixForm[-Rot1]];
    gib aus               Matritzenform
    Print["WOc2 = ", WOc2];
    gib aus
    PMW = {{Rot1[[1, 1]], Rot1[[1, 2]], Rot1[[1, 3]], WOc2[[1]]},
      {Rot1[[2, 1]], Rot1[[2, 2]], Rot1[[2, 3]], WOc2[[2]]},
      {Rot1[[3, 1]], Rot1[[3, 2]], Rot1[[3, 3]], WOc2[[3]]}, {0, 0, 0, 1}};
   ];

   ImagePlaneC2PointsWorld = Map[{#[[1]], #[[2]], zeta2, #[[3]]} &, IPC2];
                             wende an
   ImagePlaneC1PointsWorld = Map[{#[[1]], #[[2]], zeta1, #[[3]]} &, IPC1];
                             wende an

   For[i = 1, i ≤ 9, i++,
   For-Schleife
    ImagePlaneC2PointsWorld[[i, All]] = PMW.ImagePlaneC2PointsWorld[[i, All]];
                                        alle                                  alle
```

```
];

Print["ImagePlaneC1PointsWorld =",
 MatrixForm[Simplify[ImagePlaneC1PointsWorld]]];
Print["ImagePlaneC2PointsWorld =",
 MatrixForm[Simplify[ImagePlaneC2PointsWorld]]];


ImagePlaneC1PointsWorld =
 Map[{#[[1]], #[[2]], #[[3]]} &, ImagePlaneC1PointsWorld];
ImagePlaneC2PointsWorld = Map[{#[[1]], #[[2]], #[[3]]} &,
  ImagePlaneC2PointsWorld];

LinesC1 = Map[# + t (# - O2D) &, ImagePlaneC1PointsWorld];
LinesC2 = Map[# + t2 (# - WOc2) &, ImagePlaneC2PointsWorld];
Print["LinesC1 = ", LinesC1];
Print["LinesC2 = ", Simplify[LinesC2]];
solve = ConstantArray[0, {9, 1}];
For[i = 1, i ≤ 9, i++,
 solve[[i]] = Solve[{LinesC1[[i, 1]] == LinesC2[[i, 1]] &&
     LinesC1[[i, 2]] == LinesC2[[i, 2]]}, {t, t2}]
];

Print["t & t2 = ", Flatten[Simplify[solve]]];

ReconstructedPointsC1 = ConstantArray[0, {8, 3}];
ReconstructedPointsC2 = ConstantArray[0, {8, 3}];

For[i = 1, i ≤ 8, i++,
 ReconstructedPointsC1[[i]] = N[LinesC1[[i]] /. t → {t} /. solve[[i]]];
 ReconstructedPointsC2[[i]] = N[LinesC2[[i]] /. t2 → {t2} /. solve[[i]]];
];

Print["ReconstructedPointsC1 = ", ReconstructedPointsC1];
```

```
gib aus
Print["ReconstructedPointsC2 = ", ReconstructedPointsC2];
gib aus

ResizedPointsC1 = ConstantArray[0, {8, 3}];
                  konstantes Array
ResizedPointsC2 = ConstantArray[0, {8, 3}];
                  konstantes Array

scaleValueC1 =
 ObjectSize / (ReconstructedPointsC1[[2, 1, 1]] - ReconstructedPointsC1[[1, 1, 1]]);

Print["scaleValueC1 = ", scaleValueC1];
gib aus
For[i = 1, i ≤ 8, i++,
For–Schleife
 ResizedPointsC1[[i]] = ReconstructedPointsC1[[i]] * scaleValueC1;
];

Print["ResizedPointsC1 = ", ResizedPointsC1];
gib aus

scaleValueC2 =
 ObjectSize / (ReconstructedPointsC2[[2, 1, 1]] - ReconstructedPointsC2[[1, 1, 1]]);

Print["scaleValueC2 = ", scaleValueC2];
gib aus
For[i = 1, i ≤ 8, i++,
For–Schleife
 ResizedPointsC2[[i]] = ReconstructedPointsC2[[i]] * scaleValueC2;
];

Print["ResizedPointsC2 = ", ResizedPointsC2];
gib aus


GraphicPointsC1 = Map[{#[[1, 1, 1]], #[[1, 2, 1]]} &, ResizedPointsC1];
                  wende an
GraphicPointsC2 = Map[{#[[1, 1, 1]], #[[1, 2, 1]]} &, ResizedPointsC2];
                  wende an

G1 = Show[ListPlot[GraphicPointsC1[[1 ;; 8]], PlotStyle → Darker[Green]],
     zeig… listenbezogene Graphik                Darstellungsstil dunkler grün
   ListLinePlot[{GraphicPointsC1[[4, All]], GraphicPointsC1[[1, All]],
   listenbezogene Liniengraphik             alle                    alle
     GraphicPointsC1[[2, All]], GraphicPointsC1[[3, All]],
                    alle                    alle
     GraphicPointsC1[[4, All]], GraphicPointsC1[[8, All]],
                    alle                    alle
     GraphicPointsC1[[7, All]], GraphicPointsC1[[6, All]], GraphicPointsC1[[5
                    alle                    alle
       , All]], GraphicPointsC1[[8, All]]}, PlotStyle → Darker[Green]],
         alle                 alle        Darstellungsstil dunkler grün
   ListLinePlot[{GraphicPointsC1[[1, All]], GraphicPointsC1[[5, All]]},
   listenbezogene Liniengraphik             alle                    alle
```

```
    PlotStyle → Darker[Green]],
    ListLinePlot[{GraphicPointsC1[[2, All]], GraphicPointsC1[[6, All]]},
    PlotStyle → Darker[Green]],
    ListLinePlot[{GraphicPointsC1[[3, All]], GraphicPointsC1[[7, All]]},
    PlotStyle → Darker[Green]]];
G2 = Show[ListPlot[GraphicPointsC2[[1 ;; 8]], PlotStyle → Darker[Red]],
    ListLinePlot[{GraphicPointsC2[[4, All]], GraphicPointsC2[[1, All]],
      GraphicPointsC2[[2, All]], GraphicPointsC2[[3, All]],
      GraphicPointsC2[[4, All]], GraphicPointsC2[[8, All]],
      GraphicPointsC2[[7, All]], GraphicPointsC2[[6, All]], GraphicPointsC2[[5
        , All]], GraphicPointsC2[[8, All]]}, PlotStyle → Darker[Red]],
    ListLinePlot[{GraphicPointsC2[[1, All]], GraphicPointsC2[[5, All]]},
    PlotStyle → Darker[Red]],
    ListLinePlot[{GraphicPointsC2[[2, All]], GraphicPointsC2[[6, All]]},
    PlotStyle → Darker[Red]],
    ListLinePlot[{GraphicPointsC2[[3, All]], GraphicPointsC2[[7, All]]},
    PlotStyle → Darker[Red]]];
Print[Show[G1, G2, PlotRange → All]];
];
```