Recreation of Figure [17]. Objects are created for two figures. The second figure is transposed to the right side of Figure 1.

Clear["Global`*"]

(* CONSTANTS *)

pr = 2; (* plot range *)

\(
\theta = \frac{\pi}{4}; (* \text{arbitrary angle} *)
\)

\(z = \text{Exp}[\text{I}\theta]; (* \text{arbitrary vector for slanted line} *)
\)

\(x = 1.5; (* \text{offset of slanted line at} y \text{ intercept} *)
\)

\(p = .5z - x; (* \text{point on slanted line} *)
\)

\(\xi = .5z; (* \text{arbitrary small complex vector; blue arrow} *)
\)

\(a = .5;
\)

\(b = 2\text{I};
\)

(* FUNCTIONS *)

\(v[z_] := \{\text{Re} @ z, \text{Im} @ z\}; (* \text{complex to vector} *)
\)

\(f[z_] := \text{Exp}[z];
\)

(* sign given as 1 or -1 *)

\(\text{arc}[q_, r_, \theta_0_, \theta_1_, \text{sign}_:]:=\)

\(\{\text{Arrowheads} @ .02, \text{Arrow}[v @ \text{Table}[r \text{Exp}[\text{I}\theta] + q, \{\theta, \text{Range}[\theta_0, \theta_1, \text{sign} .01]\}]]\};
\)

(* The dampened sign function arrow is a bit tedious to get right. If someone knows a better way, please let me know. *)

\(\text{squigglyArrow} := \text{Module}[\{\text{points}, \text{squigglyLn}, \text{hdPt1}, \text{head}, \text{squigArrow}\},\)

\(\text{pts} = \text{Table}[\{x, (.5 / (x ^ 4)) \text{Sin}[9 \text{Pi} x]\}, \{x, \text{Range}[1, 2, .01]\}]\);

\(\text{squigglyLn} = \text{Line}[\text{pts}];\)

\(\text{hdPt1} = \{2, (.25 / 2 ^ 3) \text{Sin}[12 \text{Pi} 2]\};\)

\(\text{head} =\)

\(\{\text{Arrowheads} @ .03, \text{Arrow}[\{\text{hdPt1}, \{\text{hdPt1}[[1]] + .02, 0\}, \{\text{hdPt1}[[1]] + .18, 0\}]]\};\)

\(\text{squigArrow} = \{\text{squigglyLn}, \text{head}\};\)

\(\text{Translate}[\text{squigArrow}, \{-1, 0\}]\)
}\n
SetAttributes[{v, f}, Listable];

(* FRAMES *)

xAxis1 = Line[v @ (-pr, pr / 2)];
xAxis2 = Line[v @ (-pr / 2, pr)];
yAxis = Line[v @ (-pr I / 3, pr I)];
frame1 = {xAxis1, yAxis};
frame2 = {xAxis2, yAxis};

(* POINTS *)

ptSz = .02;

points1 = {EdgeForm@Black, FaceForm@White,
       Table[Disk[v @ pt, ptSz], {pt, {0, p, a, b, a + b}}]};

points2 = {EdgeForm@Black, FaceForm@White,
       Table[Disk[v @ pt, ptSz], {pt, Join[f @ {p, a, b, a + b}, {0}]}]};

(* CURVES *)

d = .01; (* distance between points *)

slantLnPts = Table[a z - x, {a, Range[-.8, 3, d]}];

slantLn = {Line[v @ slantLnPts]};

fSlantLnPts = f @ slantLnPts;

fStrtCrv = {Line[v @ fSlantLnPts]};

vertLnPts = Table[a + I y, {y, Range[0, pr, d]}];

vertLn = {Dashed, Line[v @ vertLnPts]};

fVertCrv = {Dashed, Line[v @ f @ vertLnPts]};

horizLnPts = Table[x + b, {x, Range[0, a, d]}];

horizLn = {Dotted, Line[v @ horizLnPts]};

fHorizCrv = {Dotted, Line[v @ f @ horizLnPts]};
curves1 = {slantLn, vertLn, horizLn};

curves2 = {fStrtCrv, fVertCrv, fHorizCrv};

(* ARROWS, LINES *)

hdSz = .02;

xiArrow = {Blue, Thick, Arrowheads @ hdSz, Arrow[v @ (p, p + \xi)]};

arcArrowl = arc[p, Abs @ \xi, 0, Arg @ \xi, 1];

arcBasel = {Dotted, Gray, Line[v @ (p, p + Abs @ \xi)]};

headPos = -IntegerPart[Length[fSlantLnPts] / 4];

slantCrvArrow = {Arrowheads @ hdSz, Arrow[v @ fSlantLnPts[[1 ;; headPos]]]};

sArrow = Scale[squigglyArrow, 1 / 2];

sArrow = Translate[sArrow, {1, 0}];

arrows1 = {xiArrow, arcArrowl, arcBasel, sArrow};

arrows2 = {slantCrvArrow};

(* TEXT *)

default = 12;

small = 8;

large = 14;

data1 = {
    {"0", 0 - .1 - .15 I, default},
    {"p", p - .15, default},
    {"\frac{p}{2}\xi", p + \xi / 2 - .1 + .05 I, small},
    {"a", a - .1 I, default},
    {"b", b - .1, default},
    {"\phi", p + 1.2 (Abs @ \xi) Exp[I Arg[\xi] / 2], small},
    {"e^2", 1.6 + .3 I, default},
    {"Figure 1. Curvature of \(e^z\) on various straight lines\nfrom
     Figure [17], Ch. 5, Needham", .75 pr, pr I / 1.5, large}
};

data2 = {
    data1[[1]],
    {"e^p", f @ p + .15 + .05 I, default},
    {"e^a", f @ a - .15 I, default},
    {"e^b", f @ b - .17 I, default}
};

text = Table[Text[Style[item[[1]], item[[3]]], v @ item[[2]], {item, #}] & @
  {data1, data2};

(* GRAPHICS *)

Graphics[
  {frame1, curves1, arrows1, points1, text[[1]]},
  Translate[{frame2, curves2, arrows2, points2, text[[2]]}, v @ 3]
  ], Background -> White, PlotRange -> {(-1.5 pr, 3 pr), (-pr, 1.5 pr)},
  ImageSize -> Large]