



Figure 26.11: A projection of center  $c$  between two lines  $\Delta$  and  $\Delta'$ .

We now turn to the issue of determining when two linear maps  $f, g$  determine the same projective map, i.e., when  $\mathbf{P}(f) = \mathbf{P}(g)$ . The following proposition gives us a complete answer.

**Proposition 26.4.** *Given two nontrivial vector spaces  $E$  and  $F$ , for any two linear maps  $f: E \rightarrow F$  and  $g: E \rightarrow F$ , we have  $\mathbf{P}(f) = \mathbf{P}(g)$  iff there is some scalar  $\lambda \in K - \{0\}$  such that  $g = \lambda f$ .*

*Proof.* If  $g = \lambda f$ , it is clear that  $\mathbf{P}(f) = \mathbf{P}(g)$ . Conversely, in order to have  $\mathbf{P}(f) = \mathbf{P}(g)$ , we must have  $\text{Ker } f = \text{Ker } g$ . If  $\text{Ker } f = \text{Ker } g = E$ , then  $f$  and  $g$  are both the null map, and this case is trivial. If  $E - \text{Ker } f \neq \emptyset$ , by taking a basis of  $\text{Im } f$  and some inverse image of this basis, we obtain a basis  $B$  of a subspace  $G$  of  $E$  such that  $E = \text{Ker } f \oplus G$ . If  $\dim(G) = 1$ , the restriction of any linear map  $f: E \rightarrow F$  to  $G$  is determined by some nonzero vector  $u \in E$  and some scalar  $\lambda \in K$ , and the proposition is obvious. Thus, assume that  $\dim(G) \geq 2$ . For any two distinct basis vectors  $u, v \in B$ , since  $\mathbf{P}(f) = \mathbf{P}(g)$ , there must be some nonzero scalars  $\lambda(u)$ ,  $\lambda(v)$ , and  $\lambda(u + v)$  such that

$$g(u) = \lambda(u)f(u), \quad g(v) = \lambda(v)f(v), \quad g(u + v) = \lambda(u + v)f(u + v).$$

Since  $f$  and  $g$  are linear, we get

$$g(u) + g(v) = \lambda(u)f(u) + \lambda(v)f(v) = \lambda(u + v)(f(u) + f(v)),$$

that is,

$$(\lambda(u + v) - \lambda(u))f(u) + (\lambda(u + v) - \lambda(v))f(v) = 0.$$

Since  $f$  is injective on  $G$  and  $u, v \in B \subseteq G$  are linearly independent,  $f(u)$  and  $f(v)$  are also linearly independent, and thus we have

$$\lambda(u + v) = \lambda(u) = \lambda(v).$$